IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:		Atty. Docket No.:	003797.00923
Joseph H. Matthews III			
Serial No.:	10/825,299	Group Art Unit:	2179
Filed:	April 16, 2004	Examiner:	Huynh, Ba
For:	User Friendly Remote System Interface	Confirmation No.:	7755

PETITION UNDER 37 C.F.R. § 1.47(a)

Mail Stop Petition
Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Sir:

Pursuant to 37 C.F.R. § 1.47(a), Applicants hereby petition the Director to accept the enclosed Declaration Under 37 C.F.R. § 1.131 (the Declaration), executed by co-inventors Joseph H. Matthews III and David Barnes, on behalf of themselves and co-inventor David Plummer, who refused to join in the Declaration. The Director is authorized to charge $200 for the petition fee (37 C.F.R. § 1.17(g)) to our Deposit Account 19-0733. The Director is authorized to charge any additional amount required or credit any overpayment to Deposit Account 19-0733.

STATEMENT OF FACTS

As demonstrated in the following statements and attached documents, co-inventor Mr. David Plummer has refused to sign the enclosed Declaration Under 37 C.F.R. § 1.131 (the Declaration).

1. The last known address of Mr. David Plummer is 25822 NE 25th St, Redmond, WA 98053.

2. On January 16, 2008, Mr. Brian Brisnehan, an attorney at Banner & Witcoff, Ltd., sent an e-mail including the Declaration, Exhibits A-C, and related materials, to Mr. Plummer at the email address, dave@xeriton.com. (Exhibit A)
3. On January 17, 2008, Mr. Plummer replied to Mr. Brisnehan's January 16 e-mail, confirming receipt of the Declaration, inquiring about certain language of the Declaration and indicating that he would not sign the Declaration in its current form. (Exhibit B).
4. On January 28, 2008, Mr. Brisnehan sent another e-mail to Mr. Plummer discussing the Declaration and requesting again that Mr. Plummer sign the Declaration. (Exhibit C).
5. On January 29, 2008, Mr. Plummer replied to Mr. Brisnehan's January 28 e-mail, indicating to Mr. Brisnehan that he would refuse to sign the Declaration. (Exhibit D).

Because Applicants have demonstrated that co-inventor David Plummer refused to join in signing the Declaration, it is respectfully submitted that this petition is proper under 37 C.F.R. § 1.47(a) and M.P.E.P. § 409.03(d) and should be granted accordingly.

Respectfully submitted,

BANNER & WITCOFF, LTD.

Date: February 15, 2008

By: /Gary D. Fedorochko/
Gary D. Fedorochko
Registration No. 35,509
For Brian J. Brisnehan
Registration No. 60,409

1100 13th Street N.W.
Suite 1200
Washington, DC 20005-4051
(202) 824-3000 (Telephone)
(202) 824-3001 (Facsimile)

EXHIBIT A

ENTERTAINMENT CENTER ARCHITECTURE

Inventors: Joseph Matthews
David Plummer
David Barnes

BACKGROUND OF THE INVENTION

1. Technical Field

The present invention relates generally to an improved operating system for a computer which provides different operating modes and features in conjunction with an entertainment setting, such as when the computer is used in conjunction with a television broadcast, movie, or a sporting event.

2. Related Art

The use of personal computers in connection with televisions and other mass media has recently become a reality. Some personal computers can now be equipped with plug-in television boards which permit viewing of a television broadcast in a window on the computer monitor. Additionally, so-called "large screen" televisions providing combined television and PC viewing capabilities have also been introduced. One such product, announced in 1996 by the Gateway Corporation, provides a personal computer system with a large viewing screen, and a wireless keyboard, remote control, and mouse.

The combination of personal computer functions with television and other media/entertainment functions creates several problems when a single viewing display

is used for both functions. For example, traditional remote controls associated with televisions do not include keys which are useful for computer applications. Providing a separate remote control for the television and one for computer functions, however, unfortunately contributes to the proliferation of remote controls in a house.

The aforementioned Gateway product provides some limited support for controlling computer functions from an integrated remote control. However, it fails to solve many problems associated with providing computer functions in a large-screen setting. Moreover, displaying conventional computer-related functions on a television screen during a television broadcast clutters up the screen and detracts from the entertainment value of the programming. Thus, merely transferring computer related user interface features from a computer display to the larger screen format are disadvantageous.

As one example, computers which use a windows-based operating system such as Windows95 provide a "taskbar" display component at the bottom part of the screen which is normally fixed on the screen. This taskbar often includes a "START" button at the lower left hand corner which, when activated by the user via a mouse or keyboard, displays a list of options for starting or "launching" applications. Displaying such a taskbar while watching a television program detracts from the program and takes up valuable display space. Moreover, when a user is sitting on a couch, requiring the user to use a mouse or keyboard to activate the start button is

inconvenient. Moreover, once the user launches an application program, it is a further inconvenience to require that the viewer use a keyboard or mouse to activate various pull-down menus and the like for the application program. Consequently, the conventional display model for launching and interacting with programs in a windowed environment cannot be readily adapted to a remote control television environment.

Another computer feature normally provided in a windowed display environment such as Windows95 is a display menu with multiple choices which can be selected by mouse or keyboard. For example, pressing the "START" button in Windows95 causes a display to be generated with options such as Programs, Settings, Find, Help, Shut Down, etc. The underlined characters in the previous list represents "accelerators" which permit the user to quickly select a desired item by pressing a single keyboard key (e.g., "p", "s", or "u") rather than scrolling through the list with a mouse or arrow key. Unfortunately, if a viewer uses a remote control device which lacks alphabetic characters, these shortcut selection techniques cannot be used. Moreover, the choices are not mutually exclusive (e.g., note that the letter "S" does not uniquely distinguish between "settings" and "shut down"; thus, the alternative "u" designator must be used to select "shut down"). Although one solution is to force the user to scroll up or down using arrow keys to select a choice, this inconveniences the user. Consequently, a need exists to improve the way in which a user navigates

through a menu list when using a remote control device to control computer functions. Moreover, providing a conventional "cascading" type menu display in an entertainment environment uses up valuable display space, causing for example a television program to be obscured with menu options.

Another problem which can occur when computer functions are implemented on a large screen such as a large-screen television display is that menu selection choices are not adequately distinguishable from the background. For example, if a television program is displayed on the screen and the user activates a menu to select a choice, the conventional menu choice highlighting techniques do not permit the selected choice to be clearly distinguished against the moving background. Consequently, a need exists to enhance highlighted menu choices in a "theater" type environment.

Another problem which occurs when a computer function is implemented on a large-screen display is that screen display resolutions which are suitable for a smaller screen display may not be suitable for a larger screen. Merely allowing the user to change the resolution size on the larger screen, as is conventional, does not adequately permit the viewer to control how large the resulting window elements are. For example, switching the screen resolution to 800 by 640 pixels for a large screen TV will result in smaller icons and menu items, thus detracting from the advantages of using a larger screen. Consequently, a need exists to permit the user to more flexibly

control the size of screen components in relation to different variables.

Yet another problem concerns how application programs behave in a windowed operating system such as Windows95. The conventional application programming model assumes that other application programs can be simultaneously executing, but does not take into account the fact that a television program or movie may be in progress on the computer display, and does not take into account the fact that in such an environment the user may have access only to a simple remote control device rather than a conventional keyboard or mouse. Consequently, a need exists to provide an enhanced mode in which application programs behave differently when a "theater mode" has been activated, in order to simplify the operation of the applications when a remote control is used.

Finally, conventional techniques for indicating to a user the contents of a folder (e.g., text files, executable programs, sound files, movies, etc.) lack sufficient detail to permit the viewer to recognize the nature of the contents of any particular item. For example, the user may merely be provided with an indication that a folder contains 3 text files, a word processor application, and a movie having a particular name. The user must actually start the movie to see whether the movie is a desirable one, and the user must either open the text files or launch a viewer application to display partial contents of the text file. In short, there is no easy way for the user to quickly determine whether any particular folder item is of interest.

SUMMARY OF THE INVENTION

The present invention relates to an operating environment for controlling a computer using a remote control device. Through providing an enhanced operating environment directed to limited input control as well as a system and method for transitioning to and from the enhanced remote operating system, the problems of the prior art are overcome.

The present invention includes a computer with graphical user interface being controlled by a remote control device as including buttons enabling specific functions which aid the operation of the computer. As it is envisioned that a user will be separated by a number of feet from the monitor, the graphical user interface is directed to provide functionality to assist in easy comprehension of options as well as enhanced controllability of the functions of the computer, enabling full control of the computer found previously in full function keyboards but now by using a minimal key remote control.

Two controls on the remote control unit are contemplated to include a start button and a menu button. The start button is contemplated to be application independent and is directed to, through its operation, allow a user to quickly pick between available applications or tasks. The menu button is contemplated to be application dependent in that the options available through its operation are dependent on the currently running application. For example, if running a spread sheet

financial program, pressing the menu button is contemplated to provide spread sheet specific menu options including applications and tasks like copy cell, copy formula, etc. If running a CD player program, pressing the menu button is contemplated to provide CD specific menu options including play, pause, stop, reverse, fast forward, etc.

Through use of the graphical user interface described herein, users are expected to become familiar with the locations of the most popular applications. The facilitate the easy selection of a popular application or task, the present invention includes symbolic accelerators which are symbols displayed on a user's display corresponding to available menu items and which have a corresponding symbol on a user's remote control. To accelerate the selection process of choosing an application or task, a user presses the appropriate button corresponding to the symbol and the computer system launches the appropriate application or task as corresponding to the depressed symbol. In the case where only number symbols are used, the symbol accelerators are referred to as numerical accelerators.

The present invention supports minimal nesting of folders and applications as navigation of multiple hierarchical trees create difficulty for users with limited input devices. To this end, the present invention includes the use of enhanced scrolling techniques to allow users to quickly select applications from listed choices without navigating through multiple levels of nested trees.

The present invention also includes highlighting selections of a selection menu so as to enable a user to determine easily from a distance the current position of a selection menu. For example, one way of highlighting the current selection is through the use of a focus frame, an enlarged frame encircling the current selection. The invention also uses alternative colors to represent the current selection, either in combination with the focus frame or without the focus frame.

To assist a user in determining what a listed application relates to, the present invention includes previewing the listed applications through a preview frame. The preview frame displays information relating to the contents of an application. To enable faster access of the preview as well as prevent unnecessary loading of system memory with an application before the application is actually requested by the user, the invention includes the system determining the identity of a current selection and retrieving the contents of a data structure which includes the preview information of the current selection item. Instead of the actually selection being loaded, a short preview of the application, or other material (a commercial for another application or product) may be displayed. Notably, the information as displayed by the preview function described herein is not required to originate with the original application. The preview information may include textual information, graphics, videos, sound clips and the like. The invention also supports interactive previews. For example, while displaying a preview of a game, a user may be allowed to at least minimally interact

with the preview to create the perception of actually playing the game embodied in the current selected application.

Further, the present invention includes the addition of a second graphical user interface as directed to current PC operating functionality (requiring full keyboards for normal operation). The operating system of the present invention supports an ability to switch between the first and second graphical user interfaces as based on the input device used to control the operating system. For example, when a user presses a key on a keyboard, the system according to embodiments of the present invention may switch to a normal PC mode of operation. Likewise, when a user presses a key on a remote control, the system according to embodiments of the present invention may switch to a theater mode of operation.

As a user switches modes of operation, the various applications are alerted to the change of mode of operation so that, when actually executed, the various applications present a GUI tailored for control via a limited input device from a distance. For example, the applications are modified to be viewed and controlled from a distance using enlarged font types, minimized nesting of options, and context sensitive function keys which change their operation based on the running application.

Other features and advantages will become evident through the following detailed description, figures and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows a computing environment as contemplated by embodiments of the present invention.

Figure 2 shows a screen image without a task bar as contemplated by embodiments of the present invention.

Figure 3 shows a screen image including a start menu but no task bar as contemplated by embodiments of the present invention.

Figure 4 shows a remote control with start and menu buttons as contemplated by embodiments of the present invention.

Figure 5 shows a flowchart of a process for assigning and displaying accelerators with applications on a menu as contemplated by embodiments of the present invention.

Figure 6 shows a flowchart of a process for accelerating selection of applications using the symbols assigned with respect to Figure 5 as contemplated by embodiments of the present invention.

Figure 7 shows a first and second related start menus as contemplated by embodiments of the present invention.

Figure 8 shows a flowchart of a process for buffering inputs of selection symbols as contemplated by embodiments of the present invention.

Figure 9 shows a flowchart of a process for switching between modes of

operation as contemplated by embodiments of the present invention.

Figure 10 shows a flowchart of a process for displaying an application specific menu as contemplated by embodiments of the present invention.

Figures 11 and 12 show application specific menus displayed with various applications.

Figures 13, 14, and 15 show start menus of a registered size on screens of various resolutions.

Figure 16 shows the hierarchical processing layers as contemplated by embodiments of the present invention.

Figure 17 shows a start menu including a focus frame as contemplated by embodiments of the present invention.

Figure 18 shows a flowchart describing a process for moving the task or application menu in relation to the focus frame.

Figures 19, 20, and 21 show various implementations of the menu scrolling technique shown in Figure 18 as contemplated by embodiments of the present invention.

Figure 22 shows a preview screen associated with a folder item as contemplated by embodiments of the present invention.

Figure 23 shows a flowchart for a process for displaying the contents of a preview screen shown in Figure 22 as contemplated by embodiments of the present

invention.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1 and the following discussions are intended to provide a brief, general description of a suitable computing environment in which the invention may be implemented. Although not required, the invention will be described in the general context of computer-executable instructions, such as program modules, being executed by a personal computer. Generally, program modules include routines, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, network PC's, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

With reference to Figure 1, an exemplary system for implementing the invention includes a general purpose computing device in the form of a conventional personal computer 20, including a processing unit 21, a system memory 22, and a

system bus 23 that couples various system components including the system memory to the processing unit 21. The system bus 23 may be any of several types of bus structures including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of bus architectures. The system memory includes read only memory (ROM) 24 and random access memory (RAM) 25. A basic input/output system 26 (BIOS), containing the basic routines that help transfer information between elements within the personal computer 20, such as during start-up, is stored in ROM 24. The personal computer 20 further includes a hard disk drive 27 for reading from and writing to a hard disk, not shown, a magnetic disk drive 28 for reading from and writing to a removable magnetic disk 29, and an optical disk drive 30 for reading from or writing to a removable optical disk 31 such as a CD ROM, or other optical media. The hard disk drive 27, magnetic disk drive 28, and optical disk drive 30 are connected to the system bus 23 by a hard disk drive interface 32, a magnetic disk drive interface 33, and an optical drive interface 34, respectively. The drives and their associated computer-readable media provide nonvolatile storage of computer readable instructions, data structures, program modules and other data for the personal computer 20. Although the exemplary environment described herein employs a hard disk, a removable magnetic disk 29 and a removable optical disk 31, it should be appreciated by those skilled in the art that other types of computer readable media which can store data that is accessible by a computer, such as magnetic

cassettes, flash memory cards, digital versatile disks, Bernoulli cartridges, random access memories (RAMs), read only memories (ROMs), and the like, may also be used in the exemplary operating environment.

A number of program modules may be stored on the hard disk, magnetic disk 29, optical disk 31, ROM 24, or RAM 25, including an operating system 35, one or more application programs 36, other program modules 37, and program data 38. A user may enter commands and information into the personal computer 20 through input devices such as a keyboard and a pointing device 42. Other input devices (not shown) may include a microphone, joystick, game pad, satellite disk, scanner or the like. These and other input devices are often connected to the processing unit 21 through a serial port interface 46 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, game port or a universal serial bus (USB). A monitor 47 or other type of display device is also connected to the system bus 23 via an interface, such as a video adapter 48. Monitor 47 includes computer monitors, LCD panels, TV monitors, and the like. In addition to the monitor, personal computers typically include other peripheral output devices (not shown), such as speakers and printers.

The personal computer 20 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 49. The remote computer 49 may be another personal computer, a server, a router, a

network PC, a peer device or other common network node, and typically includes many or all of the elements described above relative to the personal computer, although only a memory storage device 50 has been illustrated in Figure 1. The logical connections depicted in Figure 1 include a local area network (LAN) 51 and a wide area network (WAN) 52. Such networking environments are commonplace in offices, enterprise-wide computer networks, intranets, and the Internet.

When used in a LAN networking environment, the personal computer 20 is connected to the local network 51 through a network interface or adapter 53. When used in a WAN networking environment, the personal computer 20 typically includes a modem 54 or other means for establishing communications over the wide area network 52 such as the Internet. The modem 54, which may be internal or external, is connected to the system bus 23 via the serial port interface 46. In a networked environment, program modules depicted relative to the personal computer 20, or portions thereof, may be stored in the remote memory storage device. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used.

An additional input to serial port interface 46 comes from remote control 55. In this regard, the remote control 55 allows for remove operation of the computer system including the components and applications found therein.

Figure 2 shows an embodiment of a viewable display according to the present

invention. It is assumed that the image is displayed on a large screen such as a projection-type TV. When the operating system is functioning and a user is not currently engaged in an application, the operating system outputs a full screen of an image 202. An advantage of hiding all buttons, bars, icons, and the like while watching a displayed image (for example, while watching TV) is that the application model presents a unified functionally simple interface with no distractions. *As different applications are executed, embodiments of the present invention contemplate each application consuming the entire viewing area 201 so as to provide to the user the largest possible amount of information even though the user may be several feet from the monitor.*

[redacted]

Figure 3 shows display 201 after a user has requested that a "start menu" appear. A start menu, as used herein, relates to a simplified display of running and available applications. As contemplated by embodiments of the present invention, the start menu 203 is displayed upon request of a user when the user presses a dedicated start button on a remote control device or a keyboard. Once depressed, the keyboard or key pad transmits a signal to the host computer requesting a display of the start menu 203. In response, the host computer outputs a video command to display the start menu 203. As shown in Figure 3, the start menu 203 is displayed with the various tasks or applications listed therein. In particular, start menu 203 includes task

A 204, task B 205, other tasks 206, and, finally, task N 207. Alternative embodiments of the invention contemplate the start menu "creeping onto" display 201 from a variety of different directions including from the top, from the sides, etc.

Each task as shown in Figure 3 has a numeric accelerator associated with it. A numeric accelerator is a number which is has a corresponding button on remote 55 or 401. To quickly select a task, a user need only press the button corresponding to the numeric accelerator displayed with the desired task. For example, task A 204 has numeric accelerator 1, task B 205 has numeric accelerator 2, and task N 207 has numeric accelerator 0. The numeric accelerators are preferably a single digit which uniquely identify a selected item.

Figure 4 shows remote control 401 similar to that of remote control 55 as shown in Figure 1. Remote control 401 includes numeric keypad 404 and directional arrows 405, 406, 407, and 408. In particular, remote control 401 includes two additional buttons which support additional functionality for the remote control. Using only start button 402, menu button 403, and numeric keypad 404, a user is able to launch applications, select menu items, and operate application programs without the need for a mouse or alphanumeric keyboard. Further, while the terms "start" and "menu" are used repeatedly herein, the terms are intended to be used generally as related to the operations performed by them. In this regard, other names may be readily associated with these functions. For example, the name "begin" may replace

"start" and "list" may replace "menu".

Figure 5 represents a process for assigning numeric accelerators (the available keypad symbols) to applications or tasks available for execution and display on the start menu. Embodiments of the present invention contemplate each task or application being stored in a single location in memory with pointers pointing back to each task or application. These pointers are displayed on the start menu. For simplicity, these pointers are referred to herein as application identifiers. Embodiments of the present invention contemplate the application identifiers representing pointers to the actual executable tasks or applications. For example, the actual executable tasks or applications may be stored in a specific location on a hard drive with the application identifiers stored in a file or directory specifically for start menu items.

Figure 5 shows one process used to display the start menu conjunction with application identifiers. In step 501, a wait loop waits for a start button to be depressed. Once depressed, the process retrieves the stored list of application identifiers in step 502. Next, the process sorts the list of application identifiers in step 503. The sorting process may include alphabetical sorting, running v. not running sorting, user definable sorting, etc. After sorting, the process retrieves the list of available keypad symbols (e.g., numerals 0 through 9) in step 504 and assigns the available keypad symbols to the sorted application identifiers. The application

identifiers with assigned symbols are output to memory in step 506 and, finally, the system displays the combination of the application identifiers with the assigned symbols in step 507. Alternatively, the application identifiers may be initially assigned once to the symbols and, from then on, each successive display of the start menu displays the preset combinations.

Figure 6 shows numerical accelerators in operation. Upon detection of a key press in step 601, the process determines whether a start button was pressed. If the start button was not depressed, then the process responds to the key press as appropriate as represented by point A 603. If the start button was depressed, the process displays the start sequence as described in step 604. Once a keypad symbols is depressed, step 605, the process attempts to access the application or task identified by the numerical accelerator as shown in step 607. At any time, a user may press the start button again to return back to the currently active application. Finally, in step 607 the process determines whether the application was previously launched, and, if currently running, switches the current content to the running application, step 609, otherwise, it launches the application, step 608.

The tasks or applications 204 through 207 of the present invention may include a variety of user-definable tasks or applications including the option of define one task as a collection of other tasks. For example, a user (or the system) may define one task as a collection of games. Through selection of the collection of games task,

the start menu is expanded to display as selectable tasks representations of the selectable games. The representation and selection of these sub-level task are explained in greater detail with respect to Figures 7 and 8.

Figure 7 shows the start menu 203 of Figure 3 after one of the tasks or applications 204 through 207 has been selected and the selected task or application provided additional selections. The original set of tasks A through N is displayed as start menu 701 without numerical accelerators with sub-level start menu 702 juxtaposed to start menu 701. New tasks E through H now have numerical accelerators 1 through 4, respectively, which replace (and re-define) those previously associated with tasks A through N. The process by which tasks E through H are chosen is similar to the process as shown in Figure 3.

The combination of nested tasks or applications through multiple levels of start menus allows a user to input various combinations without confusion. Also, if the user remembers the keystroke combination of accelerators which accesses nested applications, the user may input the combination and quickly be transported to the end task without the delay. Embodiments of the present invention provide a process for quickly processing the input keystrokes corresponding to displayed accelerators as shown in greater detail in Figure 8.

Figures 8 and 9 show flowcharts for decision making processes resident in the system for operation on selected start menu selections. Also shown are references to a

menu. The menu relates to the pull down menus available in various graphical user interface (GUI) applications. While reference is made to the menu, it is described in greater detail herein with respect to Figures 10 through 12.

As shown in Figure 8, the process waits at step 801 until a request for a start menu is received. Once a start menu request has been received, the process determines whether a symbol (accelerator) key or keys have been pressed. If no keys have been pressed, the process displays the appropriate start menu in step 806 then returns to step 802. If a symbol key was pressed in connection with step 802, then the process loads the pressed symbol to a buffer as shown in step 803. As shown in step 804, the process attempts to match and execute the tasks selected by the depressed keys. Finally, as shown in step 805, if the final task requires the display of a start menu, then the process branches to step 806. Otherwise the last task is implemented and the system again waits at step 801 for another request for a start menu. In summary, the process shown in Figure 8 permits a user to quickly launch applications without waiting for a previous application to launch or a menu to be displayed.

To facilitate the use of the system as shown in Figure 1, multiple modes of operation may be provided. In a first "PC mode", applications operate as conventionally configured to receive input from a keyboard and other input devices. In a second "theater mode", applications receive a different type of input, where the input is geared to remote control of the computer system through a limited interface

device. In one embodiment, the primary input device for the theater mode is a remote control device such as remote control 401. Likewise, the primary input devices for the PC mode are primarily the keyboard 40 and mouse 42 of Figure 1. In particular, each application is informed of what mode the system of Figure 1 is currently operating. The user may select a particular mode by actuating keys on either the remote control or keyboard. In one embodiment, pressing any key on the remote control causes the system to switch into theater mode, while pressing any key on the keyboard causes the PC mode to be actuated. Figures 9 and 10 describe the operation and conversion between the various modes.

Figure 9 describes the operation of the system as set forth above. When the start button 401 or the menu button 402 is pressed, the process moves from step 901 to step 902. Next, the system attempts to determine the origin of the start or menu selection in step 902. If the source of the start or menu button was remote 55, then the process branches to step 903. If the source of the start or menu button was the PC input device, then the process branches to step 906. From step 903, the process determines whether the system of Figure 1 is already operating in theater mode and, if so, displays the theater mode version of the start or menu button 904. If the system of Figure 1 is not in theater mode, then the process branches to step 905 where it switches the mode to theater mode and informs the applications that the theater mode has been initiated. From step 905 the process branches to step 904. If the input device

from step 902 was a PC input device, then the process branches to step 906 where the process determines whether the system of Figure 1 is operating in PC mode. If the system is in PC mode already, the PC mode task bar and start menu are displayed. If the system is not already in PC mode, the process branches to step 907 where PC mode is initiated and the applications are informed of the conversion to PC mode. Finally, step 908 is initiated at the conclusion of step 907.

In one embodiment, the menu for the various modes is mode sensitive. Accordingly, when in different modes, different menus may be displayed for the same function (e.g., "start"). Also, the menu is application sensitive. Accordingly, when in different applications, pressing the menu button may display different menu options.

Figure 10 shows a flowchart for an operation of the menu across the different modes. From point A 603 of Figure 6, the process determines whether the menu button was the received key press (step 1001). If the menu button was not pressed, then the key press is forwarded to the currently active application (step 1005) and the current application executes the key press accordingly.

[redacted]

If a received key press in step 1001 was the menu button, then the process determines the mode of operation of the system of Figure 1 in step 1002. If running in PC mode, the system displays the standard menu mode of applications as is currently

practiced. However, if the operation of the system is in theater mode, the system detects which application is currently running. From step 1007, the process commands the currently active application to display its menu (i.e., its "theater mode" menu).

[redacted]

An example of the various menus retrieved in theater mode is shown in Figures 11 and 12. Referring to Figure 11, display 201 shows two applications currently running in theater mode. Application 1101 shows a menu with options A 1102, B 1103, C 1104, and D 1105. Application 1106 shows a menu with options W 1107, X 1108, Y 1109, and Z 1110. Figure 12 shows another application running in theater mode in which the application is the television viewing mode. In one embodiment, pressing the menu button with no applications selected causes a display of available channels such as channels 2 1201, 3 1202, 7 1203, and 9 1204. Providing a theater mode menu for each application (in addition) to traditional menus in PC mode) permits the viewer to control applications using a simple remote control device.

Current monitors 47 can display a variety of screen resolutions. For example, typical monitors can display from 640x480 lines of resolution to 1024x780 lines of resolution or more. However, when the user changes screen resolution, all items

(including menus) are changed.

Figures 13 through 15 show user interfaces 201 as scaled through the changes in resolution. In accordance with the present invention, the start menu is maintained a consistent size between resolution changes so as to maintain the ability of a user to read the start or menu at a distance. Figure 13 shows an image 1301 with start menu 1302 of a given size. Figure 14 shows image 1401 corresponding to image 1301 but at a higher resolution. In Figure 14, the start menu 1402 is maintained a current size as that of the start menu 1302. Finally, Figure 15 shows image 1501 at highest resolution with start menu 1502 at the same size of start menus 1302 and 1402.

In summary, the user can decide how big on the screen the menus should be, wherein the size is determined by a ratio of screen size to menu size. In one embodiment, all content is offered at 640 x 480 pixel resolution as the largest resolution. Two variables can be provided to the user to permit control over menu size. One factor is a "bigness" factor (e.g., 0.5 to 1.0), where 0.5 specifies that a menu should be half as big as a 1.0 sized menu. The other factor is the resolution factor (e.g., x/640). Thus, for example, a menu size is determined as:

800/640 x 400 = 500 pixels high on a large screen.

[redacted]

In accordance with one aspect of the invention, all applications are alerted to the current menu size whenever a change occurs. All fonts, buttons, and menus can thus

be scaled to the correct size based on the current screen resolution and a user-specified "bigness" factor.

Figure 16 shows the exchange of mode information between the different layers of the processing system of Figure 1. Applications (app. 1 1604, app. 2 1605 and app. 3 1606) are said to reside at the application layer 1601. The mode layer 1602 shows the differing modes available to the processing system of Figure 1. In particular, PC mode 1608 and theater mode 1609 are shown. Other modes may be added as needs exist. Through the operation of the different input devices, a user selects in which mode to operate. This selection is passed to mode selector 1610 which selects from which configuration the system of Figure 1 should be operating. The specified mode is then displayed on monitor 1603. Mode control 1611 passes to the various applications the mode in which the current system is operating. The information to be passed to the applications may be a signal indicating which mode is currently specified. Alternatively, a field may be set which is then accessed by each application in turn when executed. As the multiple applications may be running at the same time, updating all applications at once to a change in operating mode may overload the system resources of Figure 1. To this end, alerting each application in turn, through setting a shared field, allows each application to update itself in turn when accessed by the user. Accordingly, when a user switches multiple times between PC mode and theater mode, changing only the applications that need to be changed as

a user accesses them results in a savings of processing resources by not having to convert all applications when unnecessary.

As the theater mode of the present application is intended to be viewed at a distance from the user, one enhancement includes highlighting the currently selected choice on the start menu or application menu. Figure 17 shows an enhanced selection on the start menu. Tasks 1 1701, 2 1702, .. N 1704 are listed on the start menu with task 2 1702 highlighted. The highlighting may assume a variety of forms. As shown in Figure 17, the highlighting is shown by an enlarged "regional" window surrounding task 2 1702. To highlight the current selection in this way, the size of window 1705 is specified in a design table which is then accessed every time the user changes the current selection. Alternative highlighting measures may include changing the color, the size of the font of the highlighted task, etc. The purpose of the highlighting is to focus one's attention to the highlighted choice. For simplicity, the highlighted choice is referred to as a focus frame.

[redacted]

In one embodiment, enhanced menu scrolling techniques are used. Figure 18 shows a flowchart which describes moving the focus frame through a menu. From a displayed menu (step 1801), the process determines whether a keypad symbol was depressed (step 1802) and, if so, branches to point C 1803 where the system

processes the pressed keypad symbol as described in greater detail with respect to Figure 8. Otherwise, the process determines whether an arrow button was pressed in step 1804. If no arrow button was pressed, the process scheme branches to step 1805 where the system performs the action associated with the received keypress (step 1805). If an arrow button keypress was received, the process scheme locks the focus frame in its position on display 201 (step 1806) and begins to scroll the menu through the focus frame (step 1807) which becomes fixed at a particular display area. In other words, in contrast to conventional systems, the menu choices scroll through a fixed location on the screen. Finally, the process branches back to step 1802 to await another key press.

An alternative scrolling process is described in steps 1808 through 1811. In this alternative embodiment, the process moves the focus frame until near the screen border then locks the focus frame as previously described. As shown by a dashed arrow, instead of proceeding with step 1806, the process proceeds from step 1804 to step 1808 when an arrow button is depressed. Step 1808 determines whether the focus frame is one cell away from the border of display 201. If not, the process moves the focus frame in the direction of the depressed arrow button. If the focus frame is one cell away from the border of display 201, then the process locks the position of the focus frame in its position on display 201 (step 1809) and moves the menu through the focus frame. While step 1808 shows a single cell, a number of cells may

be spaced between the border and the lock position of the focus frame. An advantage in locking the focus frame at least one cell away from the border includes being able to keep the focus frame away from the distorting effects of the edge of display 201 (for example, pin cushion effects, color shifts, etc.) as well as allowing the user to see what the next item in the list of start menu is before moving to it.

An advantage of scrolling through the display choices is evident when the number of choices for a menu exceeds the number of available accelerators. Figure 19 shows an example of the processing scheme of Figure 18 including display 201 with start menu tasks F 1901, G 1902, H 1903, plurality of tasks 1905, P 1906, and Q 1907. In this example the accelerators 1-0 are applied to tasks G-P. In this regard, tasks F and Q do not have an accelerator assigned to them. By enabling the scrolling of the start menu, the assignment of the accelerators are shifted as well. For example, Figure 20 shows the embodiment of Figure 18 using steps 1806 and 1807. Once a user has pressed a down arrow button, the start menu shifts through focus frame 2004. In Figure 20, tasks G 2001 through task R 2007 are displayed. However, as with Figure 19, the accelerators only cover a few tasks, specifically tasks H 2002, task I 2003, the plurality of tasks 2005 and task Q 2006. Notably, the accelerators have shifted to provide the user with a new set of options with the new display.

By conserving the number of accelerators through the enhanced functionality of the remote control's keypad, the system eliminates the need for providing

unnecessary keys (for example, a separate scroll key) while retaining all the functionality to a user and making the transition between alternative functionality of the keys seamless to the user.

Figure 21 shows the start menu of Figure 20 exhibiting the processing scheme of Figure 18, steps 1808 through steps 1811. In particular, task G 2101 no longer has an accelerator, tasks H 2102 through Q 2105 have accelerators, now task R 2107 is visible and focus frame 2106 has moved to task Q 2105.

During operation of the menus, at times the titles of the various applications or tasks listed therein may not clearly make their contents known to the user. To more clearly represent to a user what selecting an application or task will do, embodiments of the present application incorporate a description of the application or task as described in greater detail with respect to Figure 22. Figure 22 shows a start menu 2201 with application or task identifiers relating to various games. The menu 2201 shows games with application identifiers of Monster Truck 2202, 4x4 Truck World 2203, and World Cup Soccer 2206. Block 2205 represents a plurality of applications or tasks. Focus frame 2204 surrounds 4x4 Truck World 2203, highlighting it as the current selection. While not shown, for purposes of simplicity, numeric accelerators may be used as described in greater detail above in conjunction with focus frame 2204. Alternative embodiments contemplate the accelerators used in place of focus frame 2204. Beside the name of each application or task in start menu

2201 is an icon related to each application or task. For example, the Monster Truck application 2202 includes icon 2207, the 4x4 Truck World application 2203 includes icon 2208, and the World Cup Soccer application includes icon 2209.

Focus frame 2204 highlights application 2203, focusing a user's attention to the current selection. To assist a new user, or help a veteran user remember what each application relates to, each application, task or file in menu ("items") 2201 has associated with it a list of preview parameters which preview the contents of each item to a user, once a specified menu item is highlighted. The combination of preview parameters is considered a preview data structure. The preview is shown in preview area 2212. The displayed preview can retrieve files which are not necessarily based on the content of the selected application or task. For example, an application or task may comprise only an audio file containing an interview. However, the preview of that file may include text, an image of the speaker, or a short movie showing the interviewee speaking. To this end, to facilitate the quick execution of various previews when an application is selected, the preview data structure runs quickly executing files rather than an executable associated with the application identifier shown in start menu 2201. Alternative embodiments of the present invention contemplate the preview not being associated with the content of the selected application 2203. For example, the preview files associated with the 4x4 Truck World application 2203 may relate to a commercial advertising the latest fishing supplies.

Present embodiments contemplate the preview data structure to specify at least one of a variety of multimedia presentations. For example, a preview data structure may include specified text, audio, and video previews or clips which relate to the content of the application or task. An example of a preview data structure is represented below:

preview<App Name>=

previewtext=<App Name>_preview.txt

previewaudio=<App Name>_preview.aud

previewmovie=<App Name>_preview.mpg

where <App Name> is the name of the application represented in the start menu 2201. The .txt, .aud, and .mpg file types are some of the many files available for use. Other file types may also be used including RealAudio and RealVideo files, .doc files, Gif files, JPEG files, and the like. Applied to the present example of Figure 22, where the 4x4 Truck World application 2203 is currently highlighted, the preview file associated with the application 2203 is "preview4x4 Truck World". Alternative embodiments contemplate alternative representations of the applications name when used in conjunction with the preview file data structure name. For example, the application name for "4x4 Truck World" may be concatenated to "4x4", making the

preview data structure's name "preview4x4".

The icon or movie or animation 2210 displays information relating to highlighted application 2203. Also, text 2211 provides some further descriptive text relating to highlighted application 2203. The process associated with the selection and execution of the preview information is shown in greater detail in Figure 23. Once the folder preview function of the start menu is initiated (by user designation or automatically by the system of Figure 1 every time a user selects a folder of applications) 2301, the process determines which application or task is currently selected 2302. Next, the process highlights the current selection via moving the focus frame 2204 over the current selection 2303. The process then retrieves the preview files associated with the current selection 2304 and displays or plays (as appropriate) the retrieved preview files 2305. In step 2306, the process determines whether the selection was chosen by the user for execution. If so, the process returns retrieves and executes the selected application in step 2307. If not, the process returns back to step 2302 where the new position of the current selection is determined and previewed accordingly.

Alternatively, the files relating to the preview of any task or application can be stored at a remote location. To access these files at the remote location, the file name as specified in the preview data structure described above may include the entire address of the file so that one may retrieve the remotely stored preview file. For

example, the entire address of the file may include the URL of the file as accessible over the Internet. An advantage of storing the preview file at a remote location which needs to be accessed before the preview information can be displayed includes the ability to quickly update the remotely stored file and have every user see the new preview when next accessed by each user's system. Using a remotely stored file and a URL in an application pointing to it, the provider of the preview file may additionally provide time sensitive previews (including time sensitive offers and the like) for various purposes including advertising purposes and the like.

Of course, it should be understood that the configuration of the modules discussed above is merely by way of example, and that it is contemplated that other configurations and additional (or fewer) modules could also have been implemented.

In general, it should be emphasized that the various components of embodiments of the present invention can be implemented in hardware, software or a combination thereof. In such embodiments, the various components and steps would be implemented in hardware and/or software to perform the functions of the present invention. Any presently available or future developed computer software language and/or hardware components can be employed in such embodiments of the present invention. For example, at least some of the functionality mentioned above could be implemented using the C or C++ programming language.

It is also to be appreciated and understood that the specific embodiments of

the invention described hereinbefore are merely illustrative of the general principles of the invention. Various modifications may be made by those skilled in the art consistent with the principles set forth hereinbefore.

CLAIMS

We claim:

1. A system for selecting options associated with a computer comprising:

a display device for displaying a first list of menu options and a list of unique accelerators where each accelerator is associated with one of said first list of menu options;

a receiving device for receiving from a remote control device a key press associated with one of said accelerators;

an execution device for executing a function corresponding to the one accelerator.

2. The system according to claim 1 wherein said function is an executable file.

3. The system according to claim 1, wherein said display also displays an additional option not associated with one of said list of accelerators wherein said additional option is a currently running application.

4. The system according to claim 1, wherein said display also displays an additional option not associated with one of said list of accelerators wherein all accelerators have been previously assigned to other options of said first list of options.

5. The system according to claim 1, wherein said display device further

displays a second list of options upon execution the option of said first list of options associated with said received key press wherein said list of accelerators is removed from association with said first list of options and associated with said second list of options.

6. The system according to claim 1, further comprising a storage for storing at least one additional key press associated with said accelerators in addition to said received key press,

wherein said execution device determines the association of said at least one additional key press to at least one additional option and executes said at least one additional option before said display device displays said additional option.

7. A method for selecting options associated with a computer comprising the steps of:

associating a first list of menu options and a list of unique accelerators where each accelerator is associated with one of said first list of options;

displaying the combination of said first list of options with said list of accelerators in a menu format;

receiving from a remote control device a key press corresponding to one of said list of accelerators; and,

executing a function corresponding to the one accelerator.

8. The method according to claim 7 wherein said function is an

executable file.

9. The method according to claim 7, further comprising the step of:

displaying an additional option not associated with one of said list of accelerators wherein said additional option is a currently running application.

10. The method according to claim 7, further comprising the step of:

displaying an additional option not associated with one of said list of accelerators wherein all accelerators have been previously assigned to other options of said first list of options.

11. The method according to claim 7, further comprising the steps of:

unassociating said list of accelerators from said first list of options upon execution of said function;

associating said list of accelerators with a second list of options to be displayed in response to said execution;

and displaying said second list of options with said associated list of accelerators.

12. The method according to claim 7, further comprising the step of:

storing at least one additional key press associated with said accelerators in addition to said received key press;

determining the association of said at least one additional key press to at least one additional option;

executing said at least one additional option before a display of said additional option.

13. A computer-readable medium having computer executable instructions for performing steps comprising:

associating a first list of menu options and a list of unique accelerators where each accelerator is associated with one of said first list of options;

displaying the combination of said first list of options with said list of accelerators in a menu format;

receiving a key press associated with one of said list of accelerators;

executing a function corresponding to the one accelerator.

14. The computer-readable medium of claim 13 having further computer-executable instructions wherein said function is an executable file.

15. The computer-readable medium of claim 13 having further computer-executable instructions for performing the step of:

displaying an additional option not associated with one of said list of accelerators wherein said additional option is a currently running application.

16. The computer-readable medium of claim 13 having further computer-executable instructions for performing the steps of:

displaying an additional option not associated with one of said list of accelerators wherein all accelerators have been previously assigned to other options

of said first list of options.

17. The computer-readable medium of claim 13 having further computer-executable instructions for performing the steps of:

unassociating said list of accelerators from said first list of options upon execution of said option of said first list of options associated with said received key press;

associating said list of accelerators with a second list of options to be displayed in response to said execution;

and displaying said second list of options with said associated list of accelerators.

18. The computer-readable medium of claim 13 having further computer-executable instructions for performing the steps of:

storing at least one additional key press associated with said accelerators in addition to said received key press;

determining the association of said at least one additional key press to at least one additional option;

executing said at least one additional option before a display of said additional option.

19. A system for controlling computer functions, comprising:

a remote control device comprising a dedicated start key which generates a

start signal indicating that a start menu should be displayed, a dedicated menu key which generates a menu signal indicating that an application menu should be displayed, and a plurality of numeric keys each of which generates a numeric signal indicating a numeric selection;

a computer including a receiver which receives the signals generated by the remote control device; and

a display device coupled to the computer and controlled by software executing on the computer;

wherein the remote control device transmits signals to the receiver of the computer over a wireless path; and

wherein the computer in response to receiving the start signal causes the display device to display a start menu comprising choices each having an associated numeric accelerator and, in response to receiving one of the numeric signals, launches an application program associated with one of the numeric accelerators and, in response to receiving the menu signal, causes a launched application program to display a menu providing choices pertinent to that particular application.

20. The system of claim 19, wherein the computer in response to detecting the start signal provides information to applications executing on the computer which causes them to display menu information in a different manner than if the start signal had not been received from the remote control.

21. The system of claim 19, wherein the computer inhibits the display of all taskbars, menus, and buttons until start signal is received.

22. The system of claim 19, wherein, in response to the launching of the application program, said computer removes the association of said numeric accelerator from the choices of the start menu and associates said numeric accelerators with choices of a nested menu.

23. The system of claim 19, wherein the operation of at least one of said dedicated start key, said dedicated menu key, and said numeric keys causes said computer to switch from a first mode of operation to a second mode of operation.

24. A method for controlling computer functions, comprising the steps of:

providing a remote control device comprising a dedicated start key which generates a start signal indicating that a start menu should be displayed, a dedicated menu key which generates a menu signal indicating that an application menu should be displayed, and a plurality of numeric keys each of which generates a numeric signal indicating a numeric selection;

receiving at a computer the signals generated by the remote control device over a wireless path;

in response to receiving the start signal, displaying on a display device coupled to the computer and controlled by software executing on the computer a start menu comprising choices each having an associated numeric accelerator;

in response to receiving one of the numeric signals, launching an application program associated with one of the numeric accelerators; and,

in response to receiving the menu signal, causing a launched application program to display a menu providing choices pertinent to that particular application.

25. The method of claim 24, comprising the further steps of:

in response to detecting the start signal, providing information from said computer to applications executing on the computer; and

causing said applications to display menu information in a different manner than if the start signal had not been received from the remote control device.

26. The method of claim 24, including the step of:

inhibiting the display of all taskbars, menus, and buttons until the start signal is received.

27. The method of claim 24, comprising the further steps of:

in response to the launching of the application program, removing the association of said numeric accelerator from the choices of the start menu; and,

associating said numeric accelerators with choices of a nested menu.

28. The method of claim 24, comprising the step of:

switching from a first mode of operation to a second mode of operation in response to the operation of at least one of said dedicated start key, said dedicated menu key, and said numeric keys.

29. A computer-readable medium having computer executable instructions for performing steps comprising:

receiving at a computer the signals generated by a remote control device over a wireless path, said remote control device including a dedicated start key which generates a start signal indicating that a start menu should be displayed, a dedicated menu key which generates a menu signal indicating that an application menu should be displayed, and a plurality of numeric keys each of which generates a numeric signal indicating a numeric selection;

in response to receiving the start signal, displaying on a display device coupled to the computer and controlled by software executing on the computer a start menu comprising choices each having an associated numeric accelerator;

in response to receiving one of the numeric signals, launching an application program associated with one of the numeric accelerators; and,

in response to receiving the menu signal, causing a launched application program to display a menu providing choices pertinent to that particular application.

30. The computer-readable medium of claim 29 having further computer-executable instructions for performing the steps of:

in response to detecting the start signal, providing information from said computer to applications executing on the computer; and

causing said applications to display menu information in a different manner

than if the start signal had not been received from the remote control device.

31. The computer-readable medium of claim 29 having further computer-executable instructions for performing the step of:

inhibiting the display of all taskbars, menus, and buttons until the start signal is received.

32. The computer-readable medium of claim 29 having further computer-executable instructions for performing the steps of:

in response to the launching of the application program, removing the association of said numeric accelerator from the choices of the start menu; and,

associating said numeric accelerators with choices of a nested menu.

33. The computer-readable medium of claim 29 having further computer-executable instructions for performing the step of:

switching from a first mode of operation to a second mode of operation in response to the operation of at least one of said dedicated start key, said dedicated menu key, and said numeric keys.

34. A system comprising:

a computer;

an input device coupled to the computer which receives input from a user; and

a display device coupled to the computer, wherein the display device can be operated in one of a plurality of screen resolutions;

wherein the computer controls the display device to generate menus of a particular physical size relative to the display device screen size by determining a ratio of screen size to menu size in accordance with input from the user, such that a user can control the size of the menus independently of resolution changes caused by changing to a different screen resolution.

35. The system of claim 34, wherein the computer multiplies a "bigness" factor entered by the user with a current screen resolution to determine the size of the menu.

36. A method comprising:

operating a display device coupled to said computer in one of a plurality of screen resolutions;

receiving at a computer input from a user via an input device;

controlling said display device to generate menus of a particular physical size relative to the display device screen size by determining a ratio of screen size to menu size in accordance with input from the user, such that a user can control the size of the menus independently of resolution changes caused by changing to a different screen resolution.

37. The method of claim 36, wherein the computer in said controlling step multiplies a "bigness" factor entered by the user with a current screen resolution to determine the size of the menu.

38. A computer-readable medium having computer executable instructions for performing steps comprising:

operating a display device coupled to said computer in one of a plurality of screen resolutions;

receiving at a computer input from a user via an input device;

controlling said display device to generate menus of a particular physical size relative to the display device screen size by determining a ratio of screen size to menu size in accordance with input from the user, such that a user can control the size of the menus independently of resolution changes caused by changing to a different screen resolution.

39. The computer-readable medium of claim 38 having further computer-executable instructions for performing the step of:

multiplying a "bigness" factor entered by the user with a current screen resolution to determine the size of the menu.

40. A system for highlighting a current selection comprising:

a storage for storing a list of menu options;

a display for displaying the list of menu options in response to a start or menu button, each option residing in a shape, one of said list being the current selection,

a processor for determining which of said list of menu options is the current selection, for enlarging the size of said shape, and for controlling said display to

display said enlarged shape surrounding said current selection.

41. The system according to claim 40, wherein the shape is a rectangle.

42. The system according to claim 40, where said processor alters the size of said shape by altering a registered window size applied to the current selection.

43. A method for highlighting a current selection comprising the steps of:

displaying a list of menu options in response to the reception of a signal corresponding to a start or menu button with each option residing in a shape where one of said list being the current selection,;

determining by a processor which of said list of menu options is the current selection;

enlarging the size of said shape surrounding said current selection, and,

controlling said display to display said enlarged shape surrounding said current selection.

44. The method according to claim 43, wherein the shape is a rectangle.

45. The method according to claim 43, wherein said enlarging step enlarges the size of said shape by altering a registered window size applied to the current selection.

46. A computer-readable medium having computer executable instructions for performing steps comprising:

displaying a list of menu options in response to the reception of a signal

corresponding to a start or menu button with each option residing in a shape where one of said list being the current selection;

determining by a processor which of said list of menu options is the current selection,

enlarging the size of said shape surrounding said current selection, and,

controlling said display to display said enlarged shape surrounding said current selection.

47. The computer-readable medium of claim 46 wherein said computer-executable instructions define the shape as a rectangle.

48. The computer-readable medium of claim 46 wherein said computer-executable instructions for performing the enlarging step enlarges the size of said shape by altering a registered window size applied to the current selection.

49. A system for scrolling through a list of menu options displayed on display comprising:

a display for displaying a list of menu options in response to a start or menu button where one of said menu options is a current selection,

a receiver for receiving a user command to alter the current selection with respect to the list of menu options;

a processor for controlling said list of menu options to scroll through the current location of said current selection as currently displayed on said display and for

controlling the display to display the list of menu options with the new current selection as scrolled by said processor.

50. The system of claim 49, wherein said processor moves the location of said current selection in the direction of an end of said list of menu options before moving said list of menu options through said current location.

51. The system according to claim 50, wherein said list of menu options is displayed in a number of cells, and wherein said processor scrolls said list of menu options at least one cell removed from a border of said display.

52. A method for scrolling through a list of menu options displayed on display comprising the steps of:

displaying on a display a list of menu options in response to receiving a start or menu button;

designating one of said menu options as a current selection;

receiving a user command to alter the current selection among the list of menu options;

controlling said list of menu options to scroll through the current location of said current selection as currently displayed;

controlling the display to display the list of menu options with the new current selection as scrolled by said processor.

53. The method of claim 52, comprising the further step of:

moving the location of said current selection in the direction of an end of said list of menu options before moving said list of menu options through said current location.

54. The method of claim 53, wherein said list of menu options is displayed in a number of cells, and further comprising the steps of:

fixing the location of said current selection cell; and,

scrolling said list of menu options through said current selection cell at least one cell removed from a border of said display.

55. A computer-readable medium having computer executable instructions for performing steps comprising:

displaying on a display a list of menu options in response to receiving a start or menu button;

designating one of said menu options as a current selection;

receiving a user command to alter the current selection among the list of menu options;

controlling said list of menu options to scroll through the current location of said current selection as currently displayed;

controlling the display to display the list of menu options with the new current selection as scrolled by said processor.

56. The computer-readable medium of claim 55 having further computer-

executable instructions for performing the step of:

moving the location of said current selection in the direction of an end of said list of menu options before moving said list of menu options through said current location.

57. The computer-readable medium of claim 56 having further computer-executable instructions for performing the step of:

fixing the location of said current selection cell; and,

scrolling said list of menu options through said current selection cell at least one cell removed from a border of said display.

58. A computer system for operating in one of two modes of operation comprising:

a storage storing at least one application which is operable in a first and second mode, wherein a displayed menu of said application in said first mode includes generic menu options and a displayed menu of said application in said second mode includes application specific menu options;

a display for displaying said menu options;

a receiving device for receiving input from a user comprising a request for a menu;

a processor for alerting said application in which mode said processor is currently operating, for retrieving menu options appropriate for at least one of said first and second operating modes, and for controlling said display to display said generic menu options or said application specific menu options in accordance with the current operating mode.

59. The system according to claim 58, wherein said first mode is a conventional PC mode and said second mode is a theater mode.

60. The system according to claim 58, wherein said receiving device determines from what user input device said user input is received and switches the operating mode based on origin of said user input.

61. The system according to claim 60, wherein the use of a keyboard or mouse switches the system to operate in said first mode and wherein the use of a remote control device switches the system to operating in said second mode.

62. A method for operating a computer system in one of two modes comprising the steps of:

storing at least one application which is operable in a first and second mode;

receiving input from a user comprising a request for a menu from said at least one application;

alerting said application of said computer system operating in said first mode or said second mode;

retrieving from said application a menu; and,

displaying said menu, wherein, when in said first mode, said menu includes generic menu options and, when in said second mode, said menu includes application specific menu options.

63. The method system according to claim 62, wherein said first mode is a conventional PC mode and said second mode is a theater mode.

64. The method according to claim 62, further comprising the steps of:

determining from what user input device said user input is received; and,

switching the operating mode based on origin of said user input.

65. The method according to claim 64, wherein the use of a keyboard or

mouse switches the system to operate in said first mode and wherein the use of a remote control device switches the system to operating in said second mode.

66. A computer-readable medium having computer executable instructions for performing steps comprising:

storing at least one application which is operable in a first and second mode;

receiving input from a user comprising a request for a menu from said at least one application;

alerting said application of a computer system operating in said first mode or said second mode;

retrieving from said application a menu; and,

displaying said menu, wherein, when in said first mode, said menu includes generic menu options and, when in said second mode, said menu includes application specific menu options.

67. The computer-readable medium of claim 66 having further computer-executable instructions wherein said first mode is a conventional PC mode and said second mode is a theater mode.

68. The computer-readable medium of claim 66 having further computer-executable instructions for performing the steps of:

determining from what user input device said user input is received; and,

switching the operating mode based on origin of said user input.

69. The computer-readable medium of claim 68 having further computer-executable instructions wherein the use of a keyboard or mouse switches the system to operate in said first mode and wherein the use of a remote control device switches the system to operating in said second mode.

70. A computer system comprising:

a storage for storing a list of menu options corresponding to functions and for storing a description of said menu options;

a processor for determining which of said menu options is currently selected by a user, for retrieving said description of said currently selected menu option, and providing the description to the user, wherein said description is stored apart from said function.

71. The computer system of claim 70, wherein said description includes image files.

72. The computer system of claim 70, wherein said description includes movie files.

73. The computer system of claim 70, wherein said description includes sound files.

74. The computer system of claim 70, wherein said description includes text files.

75. The computer system of claim 70, wherein said processor retrieves a data structure for each menu option which provides the identification information for a file containing said description.

76. A method of previewing menu options including the steps of:

storing a list of menu options corresponding to functions;

storing a description of said menu options,

determining which of said menu options is currently selected by a user;

retrieving said description of said currently selected menu option; and,

providing the description to the user, wherein said description is stored apart from said function.

76. The method according to claim 75, wherein said description includes image files.

77. The method according to claim 75, wherein said description includes movie files.

78. The method according to claim 75, wherein said description includes sound files.

79. The method according to claim 75, wherein said description includes text files.

80. The method according to claim 75, further including the step of:

retrieving a data structure for each menu option which provides the

identification information for a file containing said description.

81. A computer-readable medium having computer executable instructions for performing steps comprising:

storing a list of menu options corresponding to functions;

storing a description of said menu options;

determining which of said menu options is currently selected by a user;

retrieving said description of said currently selected menu option; and,

providing the description to the user, wherein said description is stored apart from said function.

82. The computer-readable medium of claim 81 having further computer-executable instructions wherein said description includes image files.

83. The computer-readable medium of claim 81 having further computer-executable instructions wherein said description includes movie files.

84. The computer-readable medium of claim 81 having further computer-executable instructions wherein said description includes sound files.

85. The computer-readable medium of claim 81 having further computer-executable instructions wherein said description includes text files.

86. The computer-readable medium of claim 81 having further computer-executable instructions that perform the step of:

retrieving a data structure for each menu option which provides the

identification information for a file containing said description.

ABSTRACT

A system, method, and medium for enhancing a GUI and viewing environment for a computer user is disclosed. Enabling a user to fully navigate and operate a computer through the use of a remote control device, the present invention includes at least a PC mode of operation and a theater mode of operation, in which a "theater" mode is designed for use with the remote device. While operating in theater mode, received function key signals are interpreted by the computer system to operate various functions which have been modified to accommodate remote viewing and control. To enable the remote device to fully control the applications executable on the computer, the disclosed process notifies the various applications of the current operating mode.

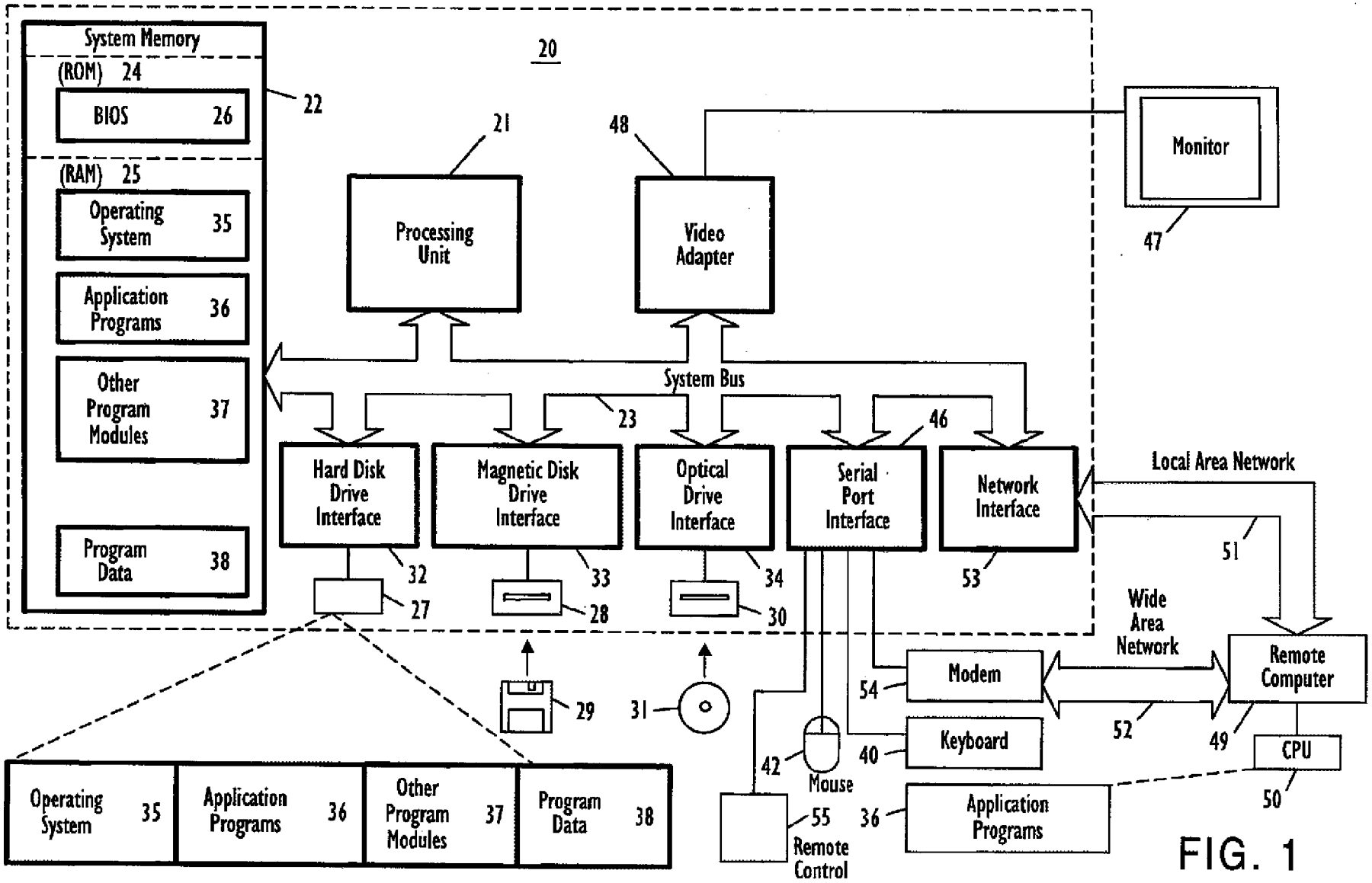

System Memory
(ROM) 24
BIOS 26
(RAM) 25
Operating System 35
Application Programs 36
Other Program Modules 37
Program Data 38
22
20
21
Processing Unit
48
Video Adapter
Monitor
47
System Bus
23
46
Hard Disk Drive Interface
Magnetic Disk Drive Interface
Optical Drive Interface
Serial Port Interface
Network Interface
Local Area Network
51
53
32
33
34
27
28
30
Wide Area Network
Remote Computer
29
31
54
Modem
52
49
CPU
50
42
Mouse
40
Keyboard
55
Remote Control
36
Application Programs
Operating System 35
Application Programs 36
Other Program Modules 37
Program Data 38

FIG. 1

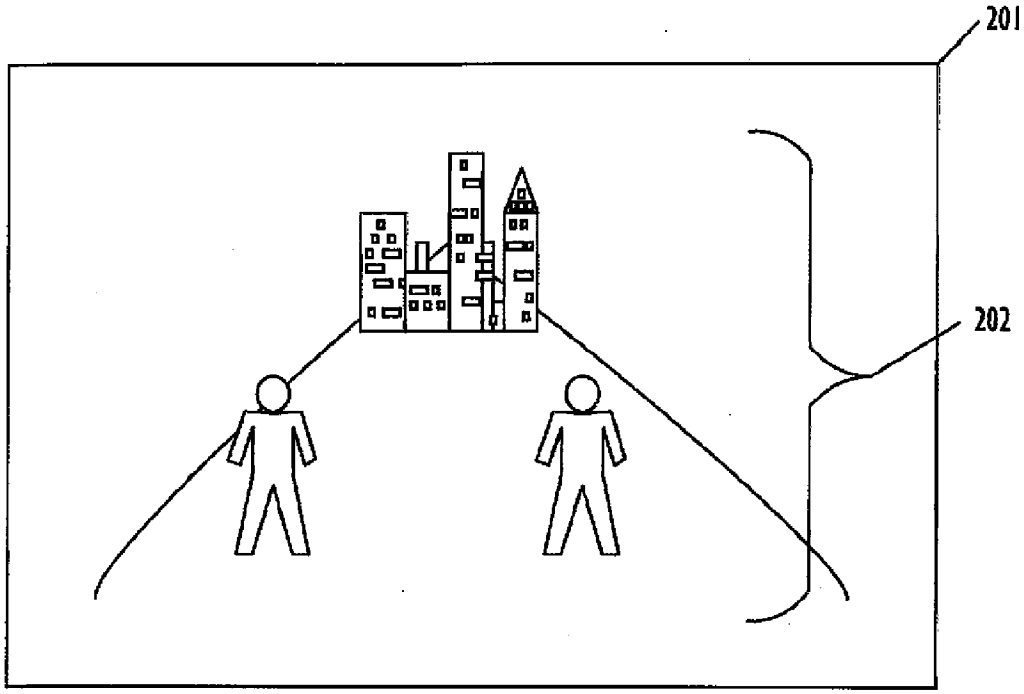


FIG. 2

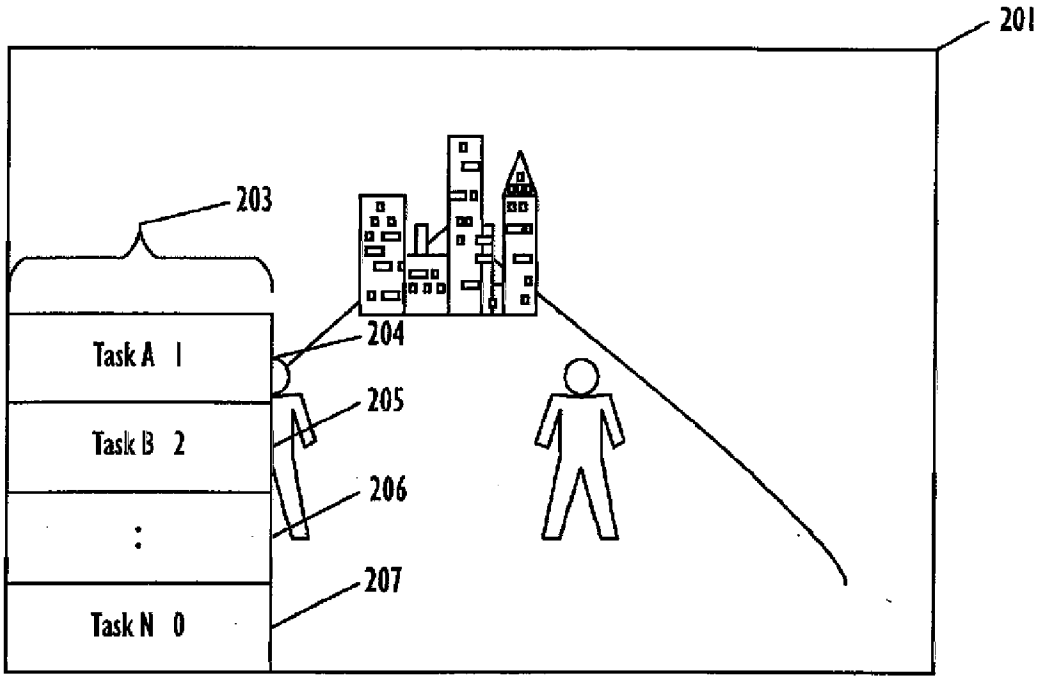


FIG. 3

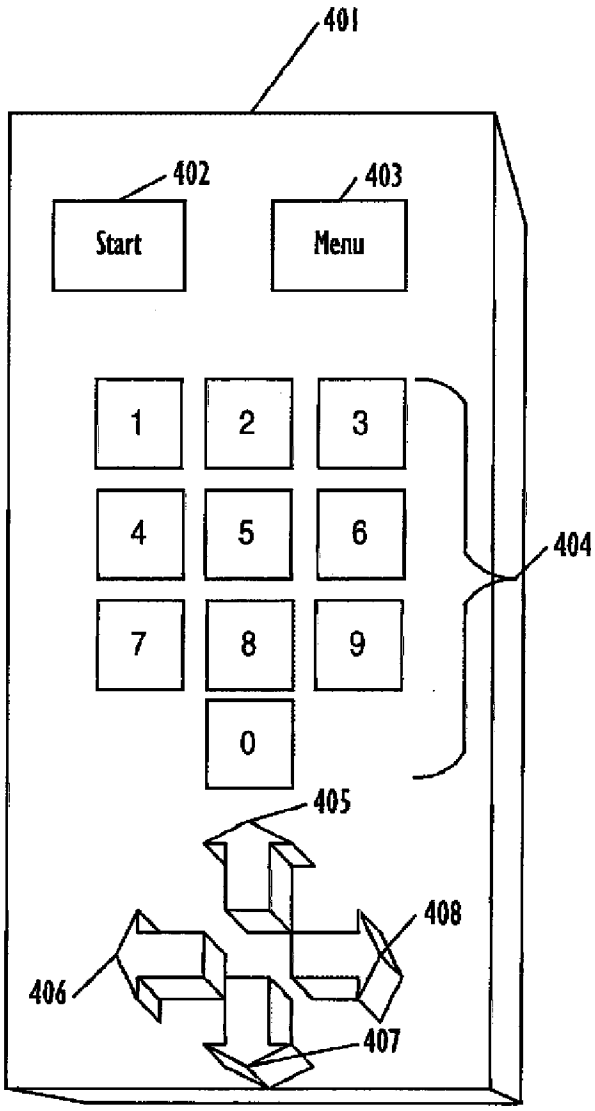


FIG. 4

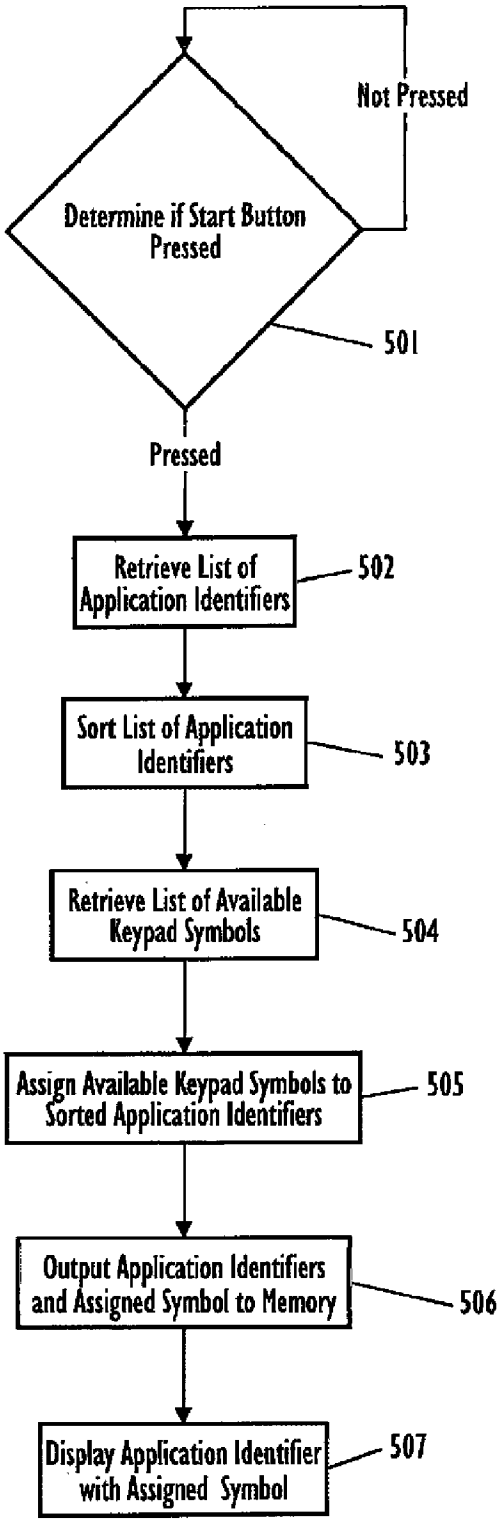

Not Pressed
Determine if Start Button Pressed
501
Pressed
Retrieve List of Application Identifiers
502
Sort List of Application Identifiers
503
Retrieve List of Available Keypad Symbols
504
Assign Available Keypad Symbols to Sorted Application Identifiers
505
Output Application Identifiers and Assigned Symbol to Memory
506
Display Application Identifier with Assigned Symbol
507

FIG. 5

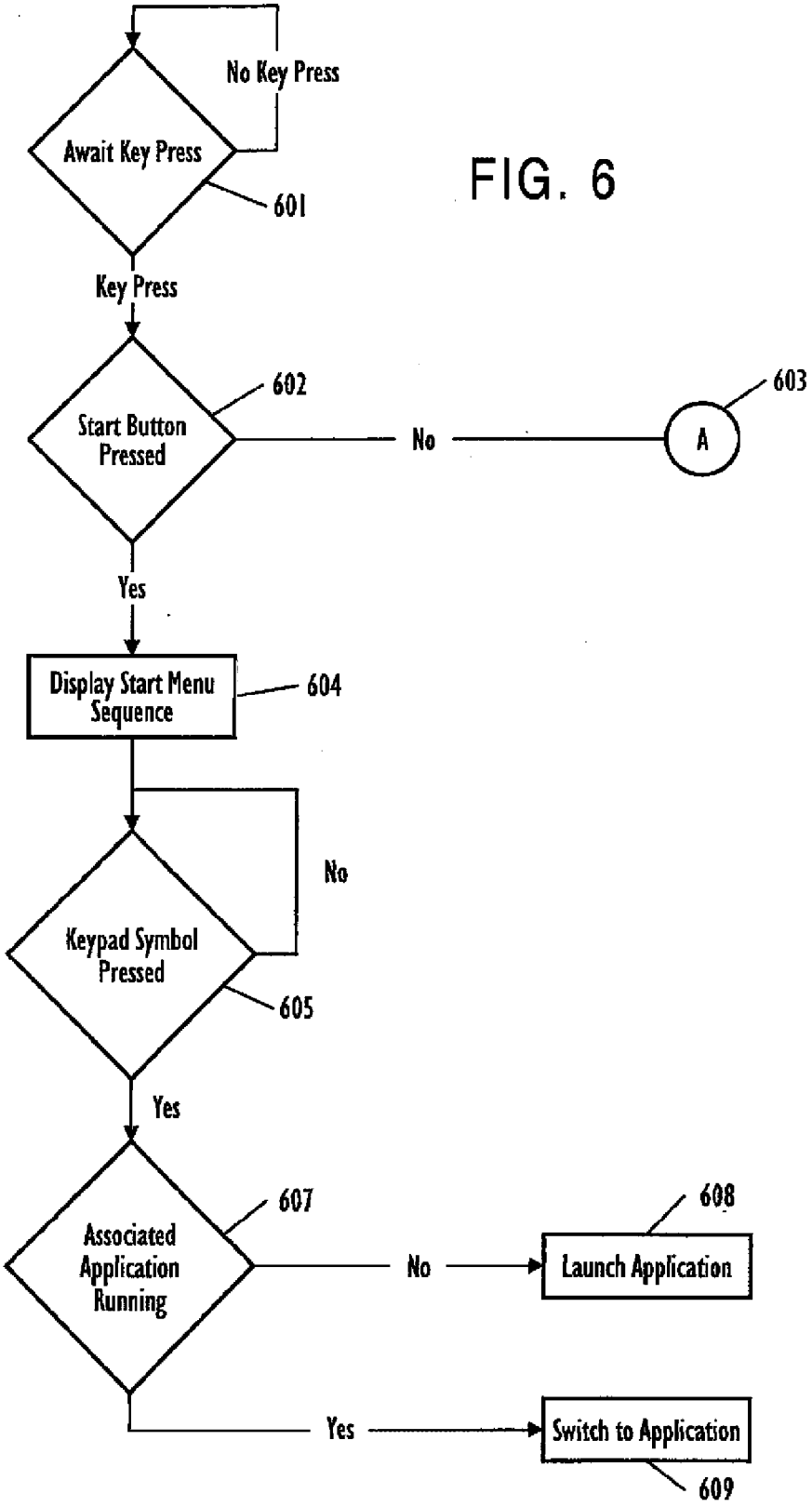
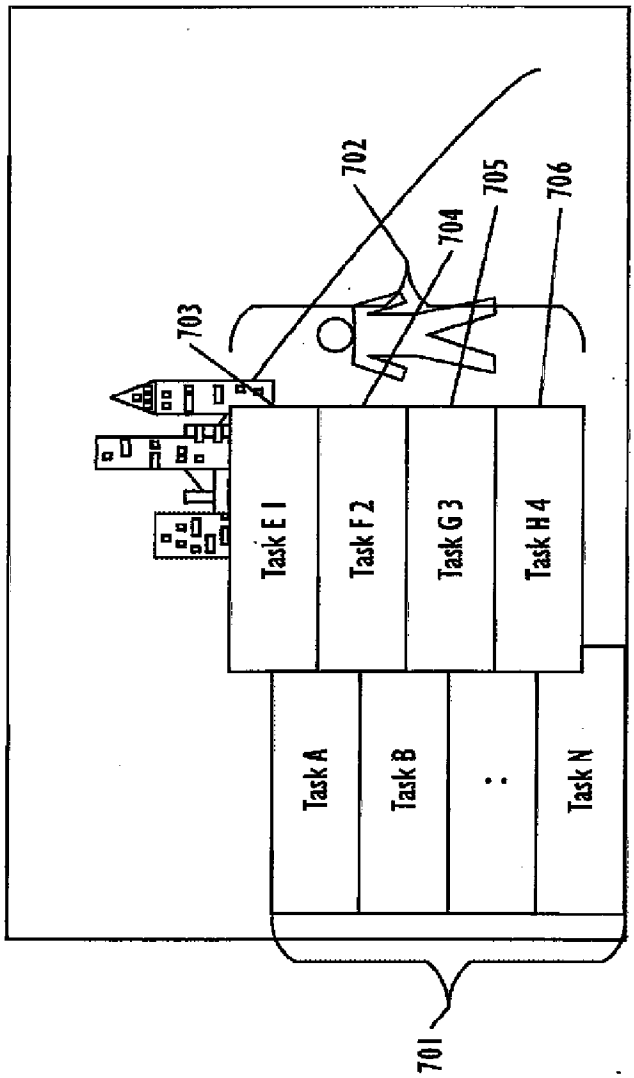

No Key Press
Await Key Press
601
Key Press
602
Start Button Pressed
No
603
A
Yes
Display Start Menu Sequence
604
No
Keypad Symbol Pressed
605
Yes
607
Associated Application Running
No
608
Launch Application
Yes
Switch to Application
609

FIG. 6

FIG. 7


701
702
703
704
705
706
Task A
Task B
:
Task N
Task E 1
Task F 2
Task G 3
Task H 4

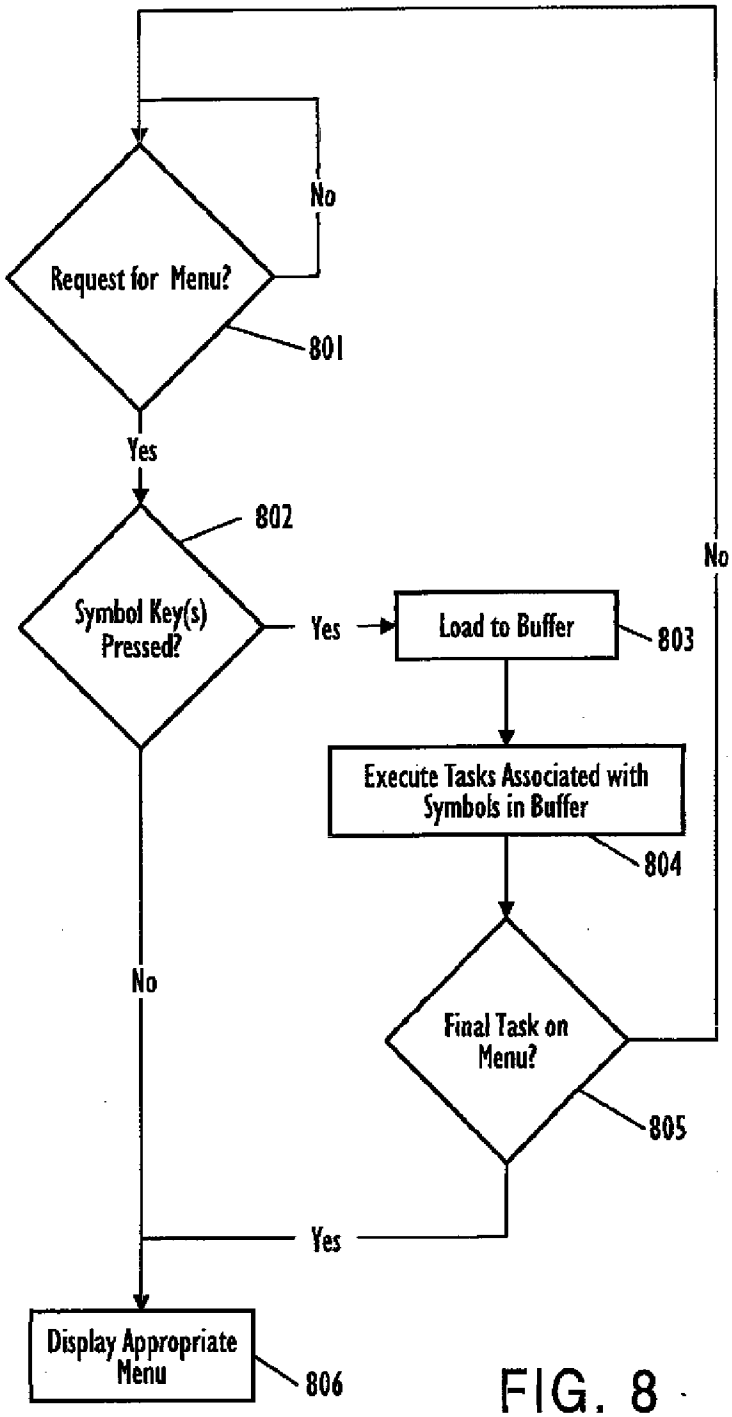

Request for Menu?
No
801
Yes
802
Symbol Key(s) Pressed?
Yes
Load to Buffer
803
Execute Tasks Associated with Symbols in Buffer
804
Final Task on Menu?
805
No
No
Yes
Display Appropriate Menu
806

FIG. 8

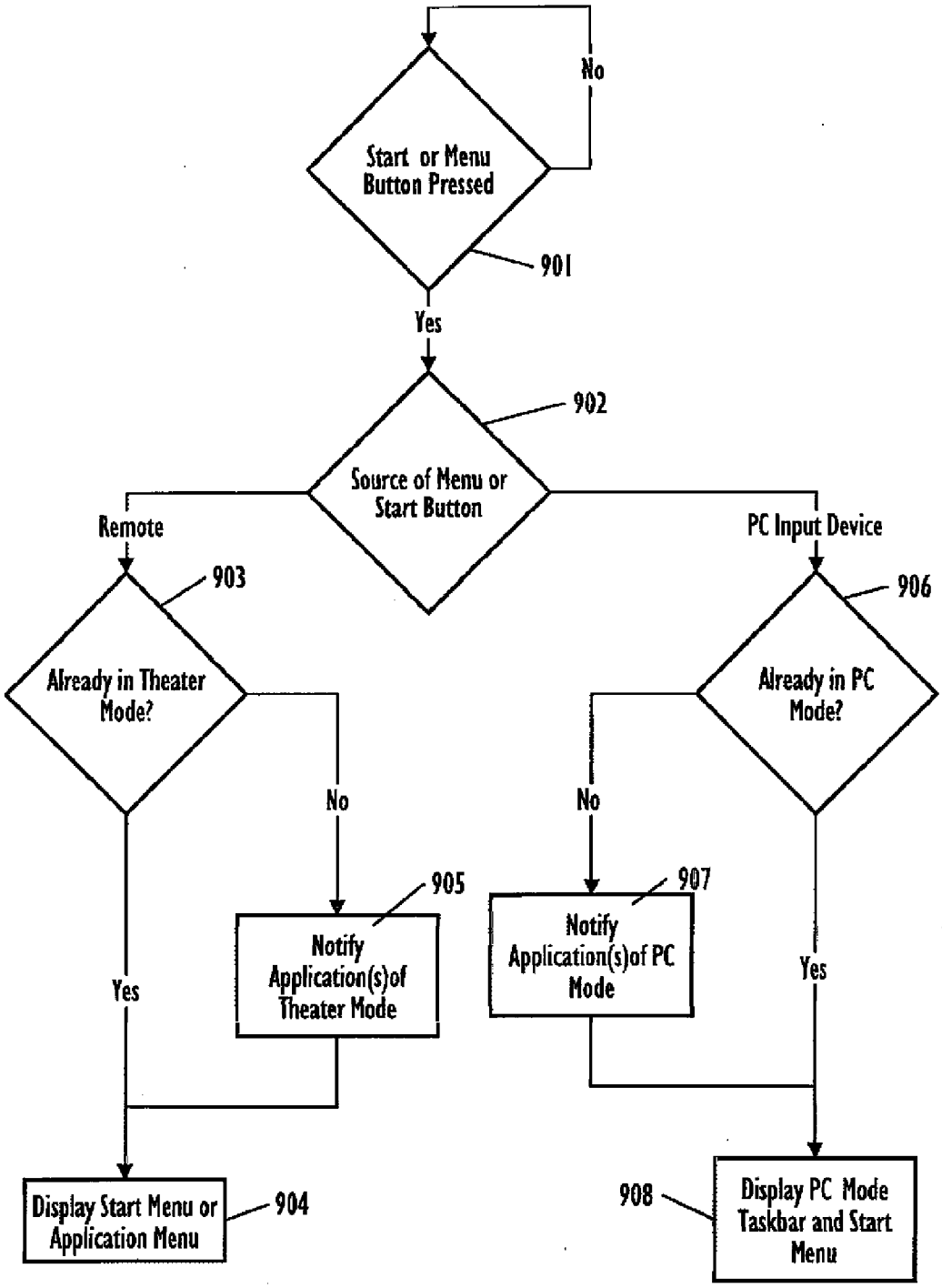

Start or Menu Button Pressed
No
901
Yes
902
Source of Menu or Start Button
Remote
PC Input Device
903
Already in Theater Mode?
No
905
Notify Application(s)of Theater Mode
Yes
906
Already in PC Mode?
No
907
Notify Application(s)of PC Mode
Yes
Display Start Menu or Application Menu
904
908
Display PC Mode Taskbar and Start Menu

FIG. 9

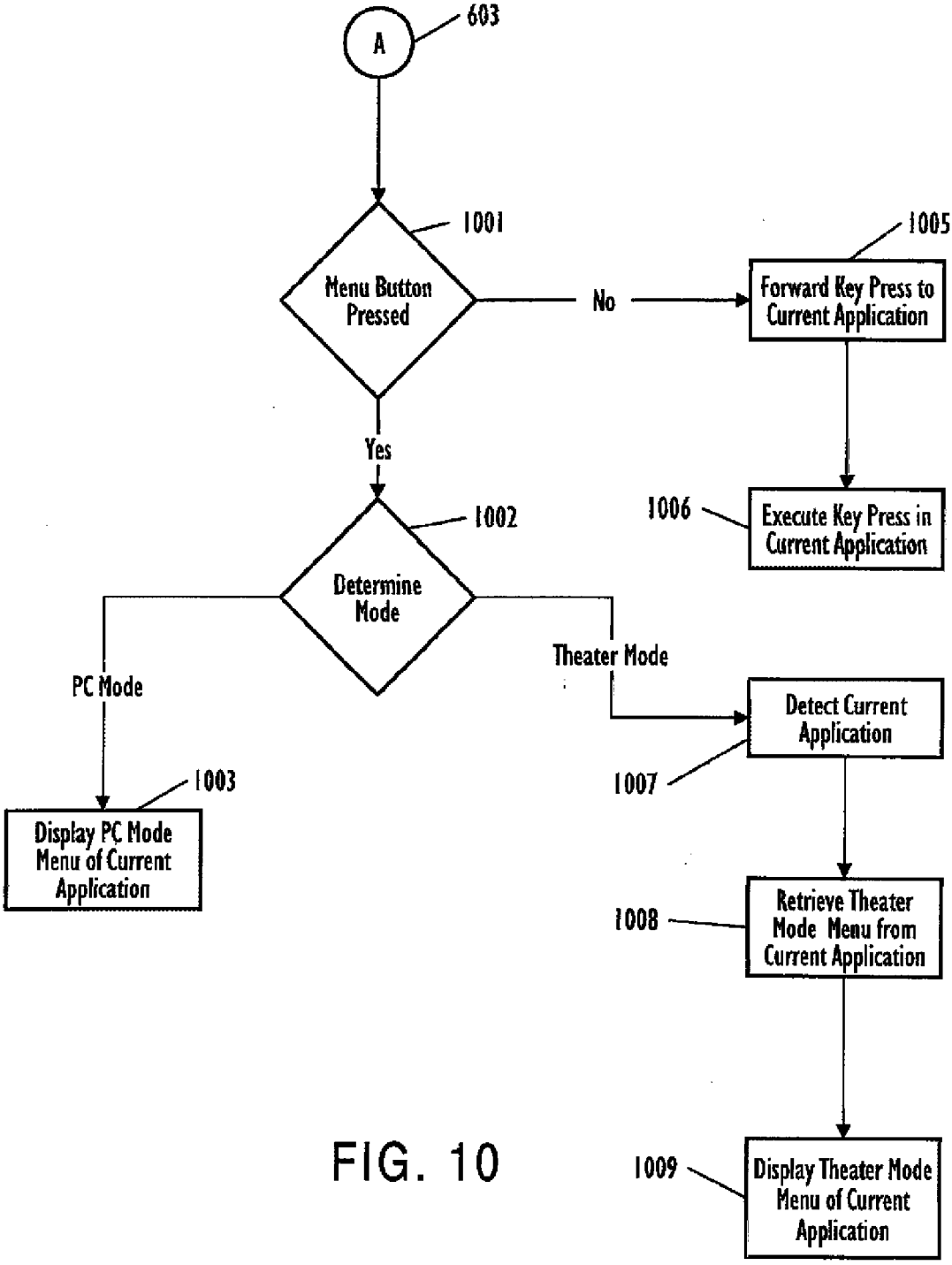

A
603
1001
Menu Button Pressed
No
1005
Forward Key Press to Current Application
Yes
1006
Execute Key Press in Current Application
1002
Determine Mode
PC Mode
Theater Mode
Detect Current Application
1007
1003
Display PC Mode Menu of Current Application
1008
Retrieve Theater Mode Menu from Current Application
1009
Display Theater Mode Menu of Current Application

FIG. 10

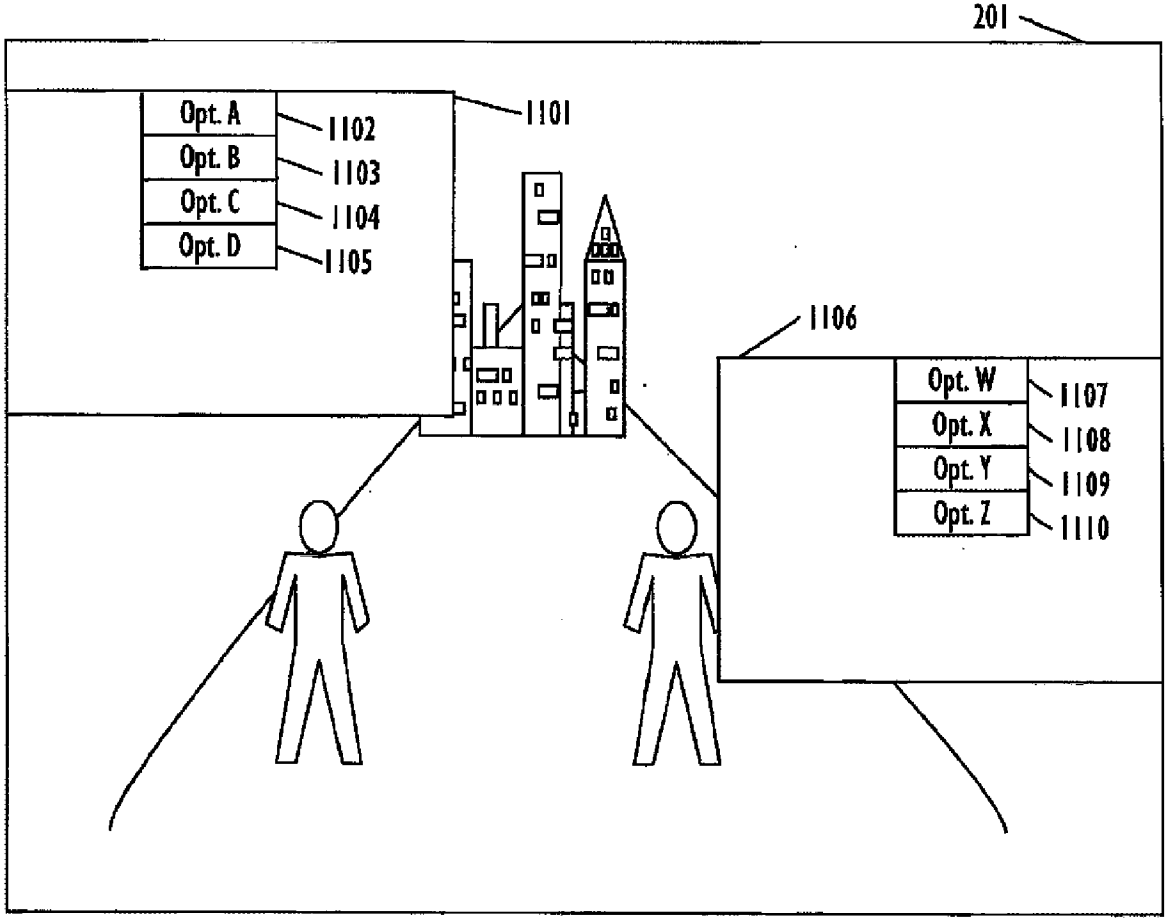

201
Opt. A
Opt. B
Opt. C
Opt. D
1101
1102
1103
1104
1105
1106
Opt. W
Opt. X
Opt. Y
Opt. Z
1107
1108
1109
1110

FIG. 11

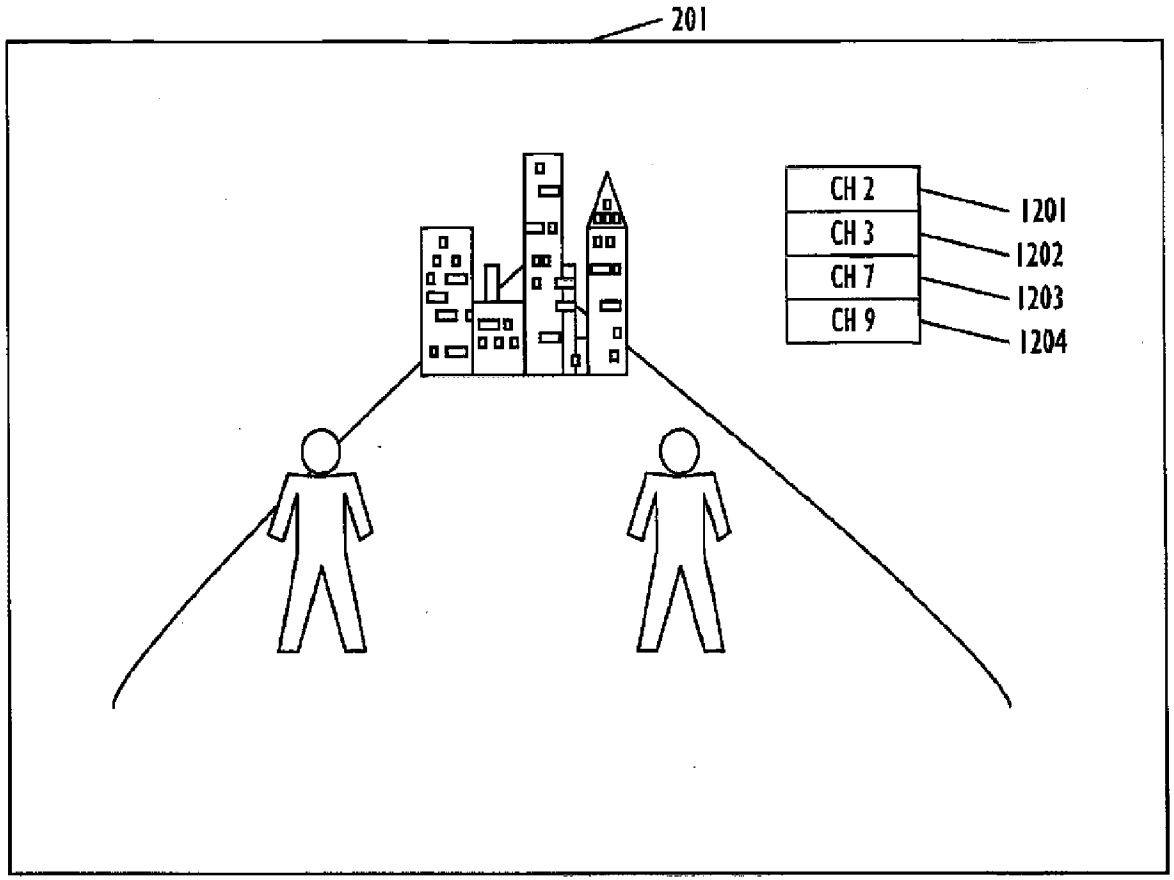


FIG. 12

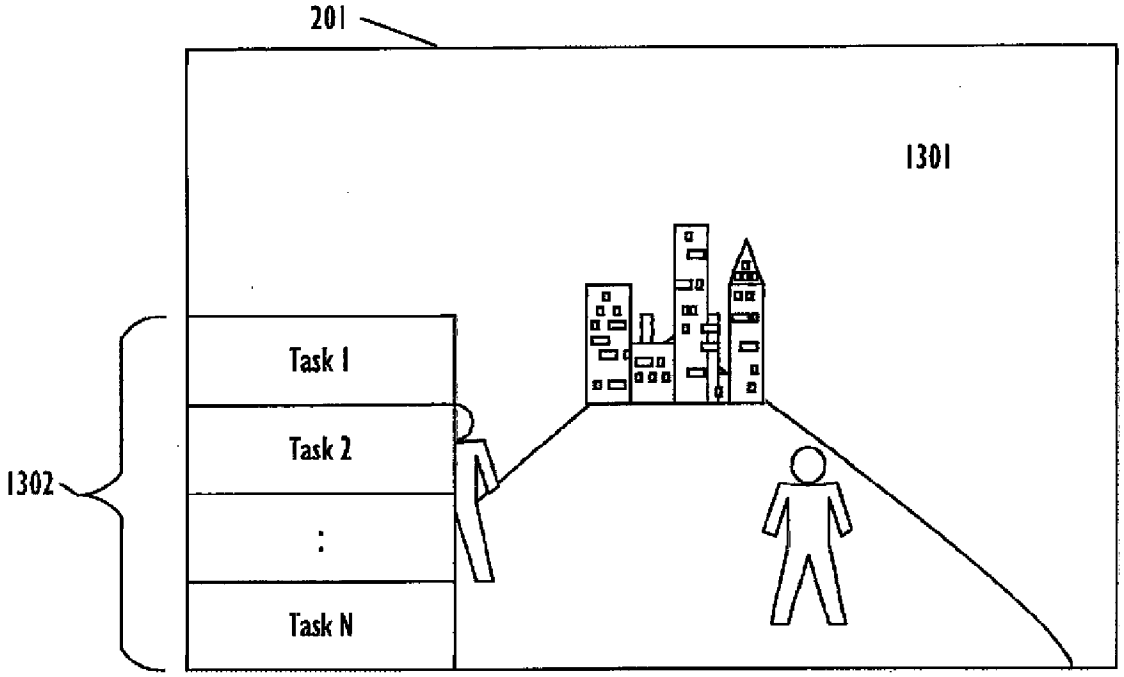

201
1301
Task 1
Task 2
:
Task N
1302

FIG. 13

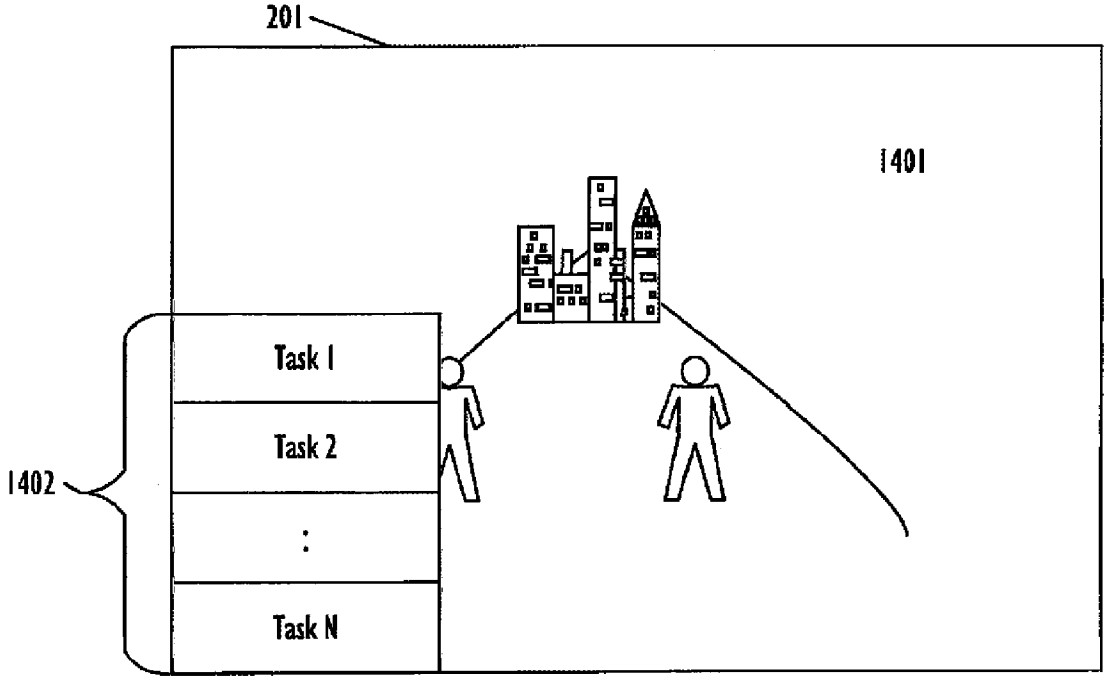

201
1401
Task 1
Task 2
:
Task N
1402

FIG. 14

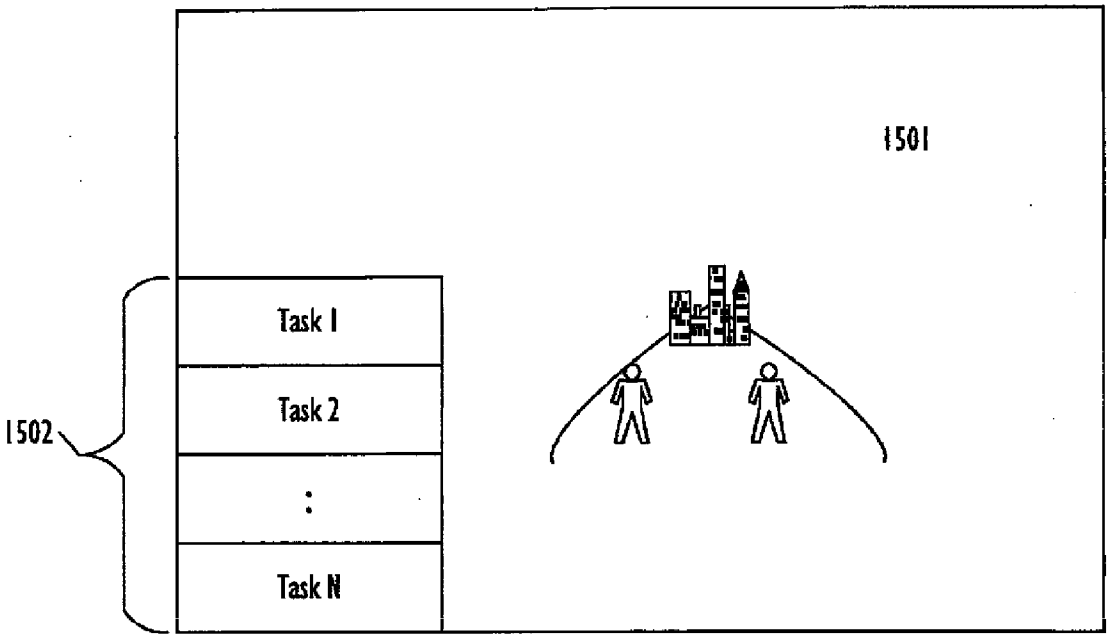

1501
Task 1
Task 2
:
Task N
1502

FIG. 15

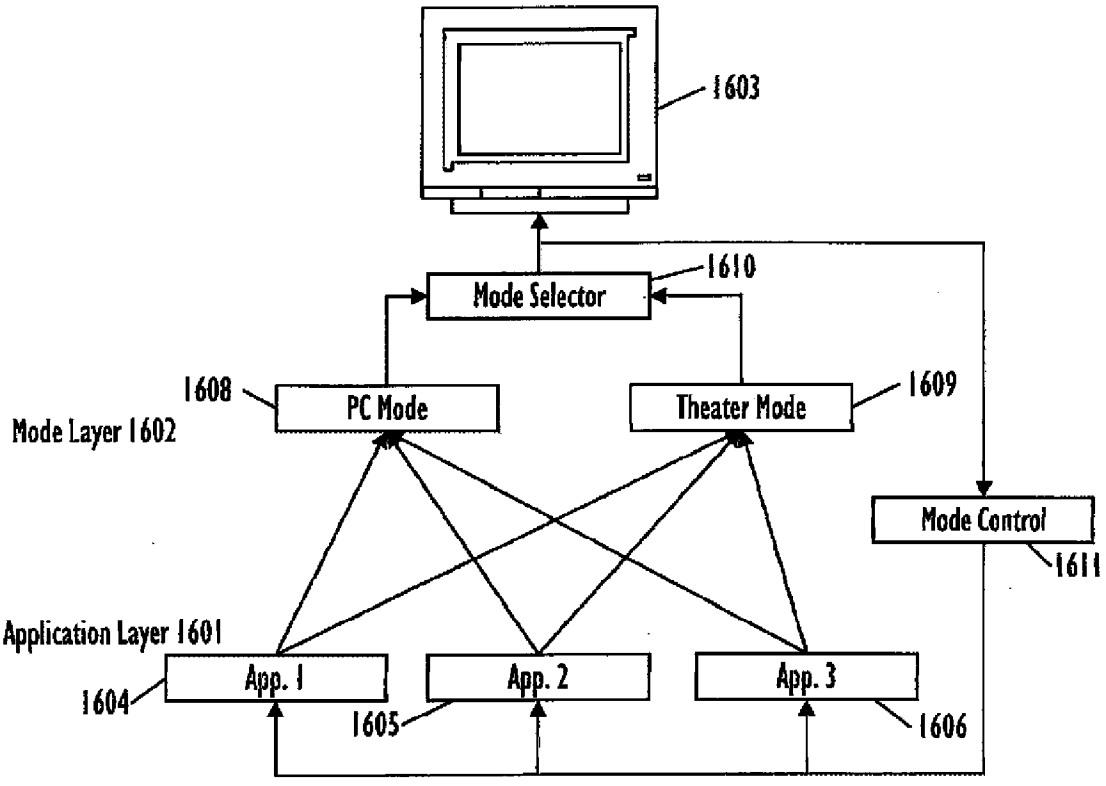

1603
1610
Mode Selector
1608
PC Mode
Theater Mode
1609
Mode Layer 1602
Mode Control
1611
Application Layer 1601
App. 1
App. 2
App. 3
1604
1605
1606

FIG. 16

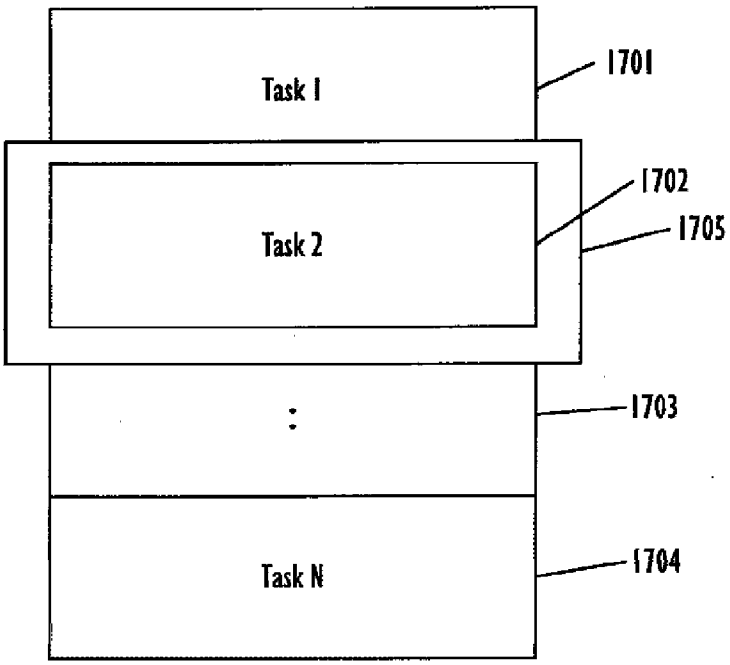


FIG. 17

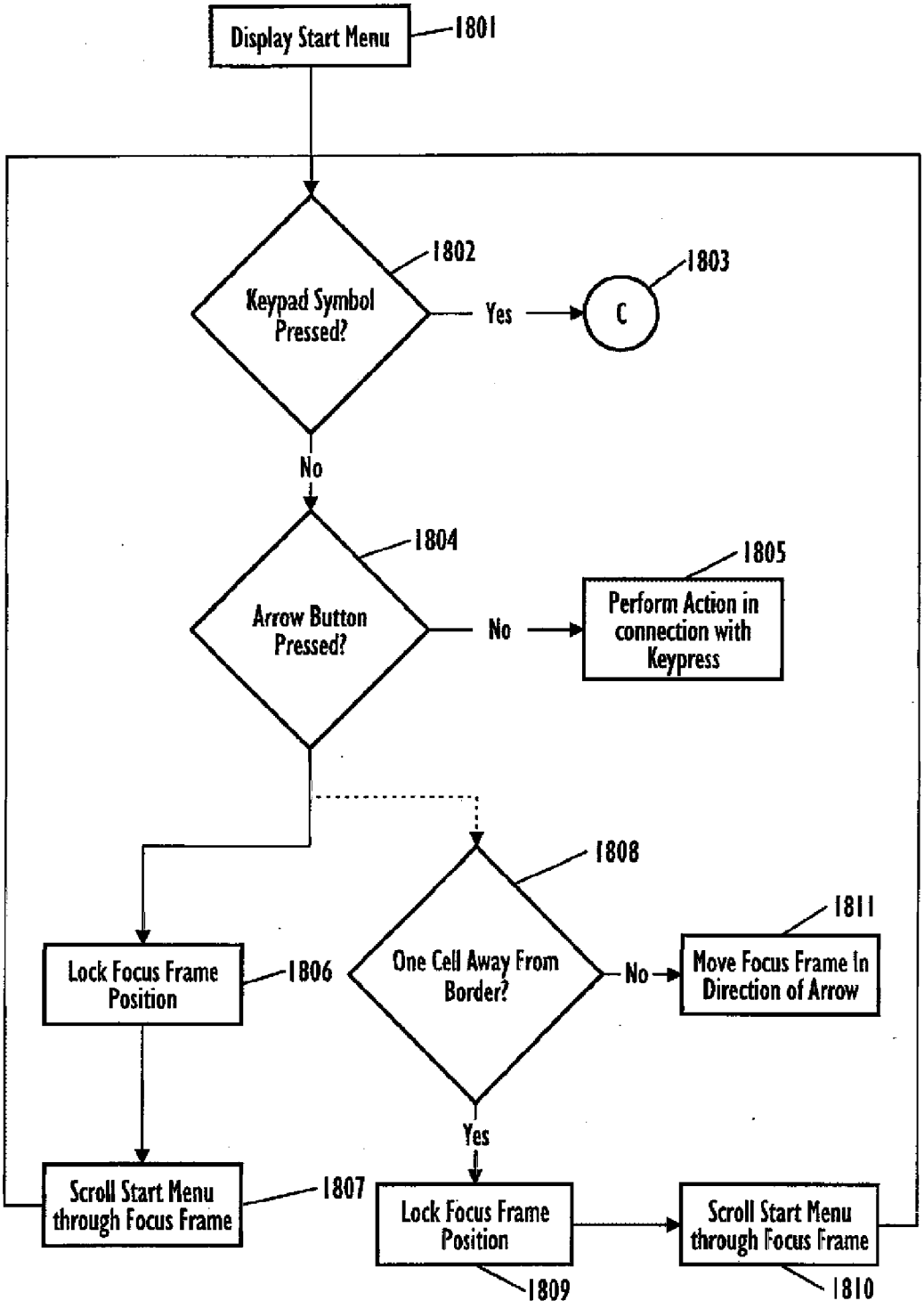

Display Start Menu
1801
Keypad Symbol Pressed?
1802
Yes
C
1803
No
Arrow Button Pressed?
1804
No
Perform Action in connection with Keypress
1805
One Cell Away From Border?
1808
No
Move Focus Frame In Direction of Arrow
1811
Lock Focus Frame Position
1806
Yes
Scroll Start Menu through Focus Frame
1807
Lock Focus Frame Position
1809
Scroll Start Menu through Focus Frame
1810

FIG. 18

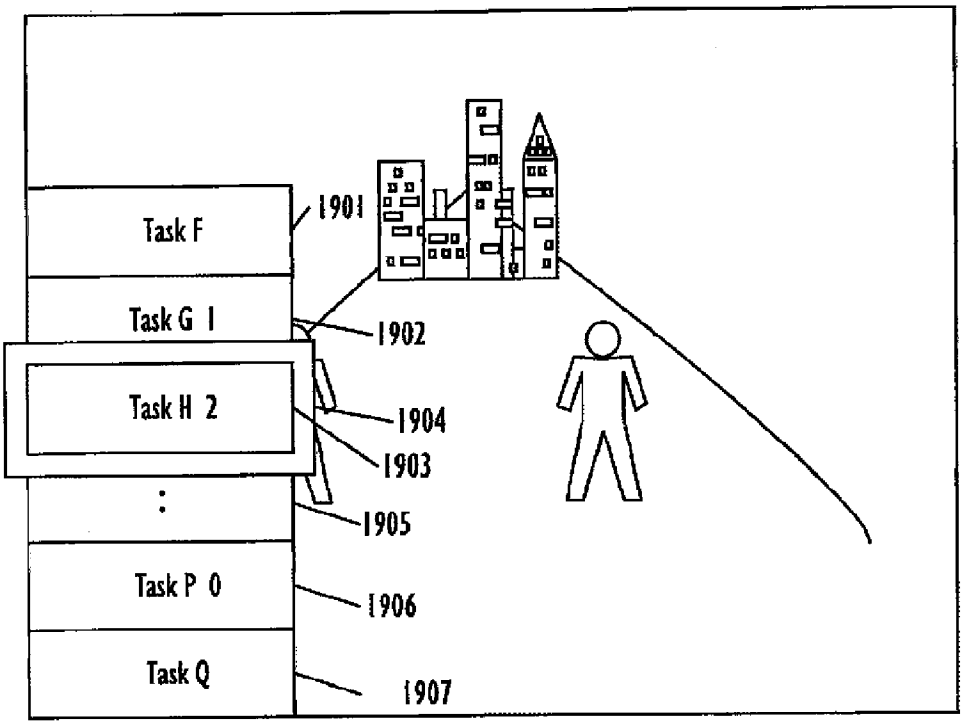


FIG. 19

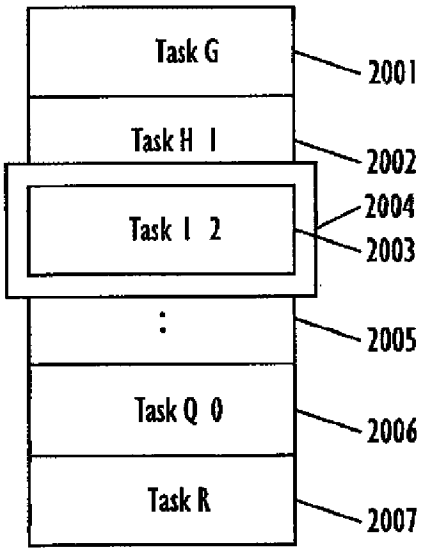


FIG. 20

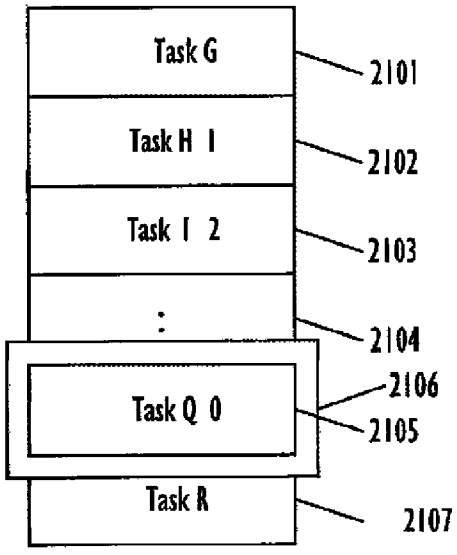


FIG. 21

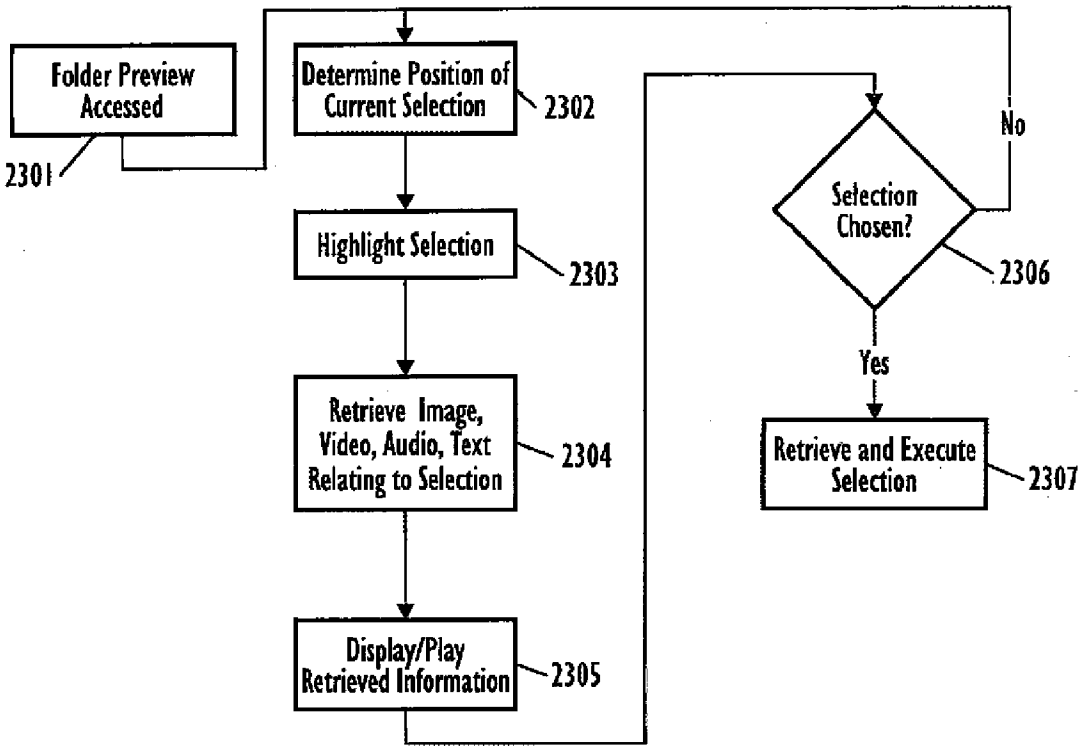

Folder Preview Accessed
2301
Determine Position of Current Selection
2302
Highlight Selection
2303
Retrieve Image, Video, Audio, Text Relating to Selection
2304
Display/Play Retrieved Information
2305
Selection Chosen?
2306
No
Yes
Retrieve and Execute Selection
2307

FIG. 23

EXHIBIT B

File Copy

LAW OFFICES

BANNER & WITCOFF, LTD.

1001 G STREET, N.W.
WASHINGTON, D.C. 20001-4597
TELEPHONE (202) 508-9100
FACSIMILE (202) 508-9299

BOSTON OFFICE
75 STATE STREET
BOSTON, MASSACHUSETTS 02109-1807
TELEPHONE (617) 345-9100
FACSIMILE (617) 345-9111

CHICAGO OFFICE
TEN SOUTH WACKER DRIVE
CHICAGO, ILLINOIS 60606-7407
TELEPHONE (312) 715-1000
FACSIMILE (312) 715-1234

[redacted]

Microsoft Patent Group Docketing Department
Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052

VIA FEDERAL EXPRESS

Re: Draft Patent Application
Entitled: Entertainment Center Architecture
Your Reference No: MS 97172
Our Reference No: 03797.09780

Re: Draft Patent Application
Entitled: Hardware Enabled Mode Switching
Your Reference No: MS 97192
Our Reference No: 03797.09782

Re: Draft Patent Application
Entitled: Previews In Folder View
Your Reference No: MS 100660
Our Reference No: 03797.10393

Enclosed is a disk copy and four paper copies of the above identified patent applications (combined into a single disclosure). The enclosed is identical to the fax copy sent yesterday [redacted]

Separate claim sets will be filed for each application. The draft claims break out as follows:

Ecenter (97172)	Claims 1-57;
Hardware Enabled Mode Switching (97192)	Claims 58-69; and,
Previews in Folder Mode (100660)	Claims 70-86.

We ask that the inventors review the application in detail to ensure that the description of the invention is accurate and includes all of the variations that they envision.

In reviewing the application, we ask that the inventors satisfy themselves that the best mode for practicing the invention is disclosed, including any changes conceived since the invention disclosure was prepared and since the discussion of [redacted] In addition, we ask

the inventors to confirm that the disclosure would enable one skilled in the art to make and use the invention. The disclosure should also include all of the significant differences between the present invention and the prior art.

We request all comments from the inventors be forwarded to us no later than [redacted]

We would like to file the applications in the U.S. Patent and Trademark Office on [redacted]

Very truly yours,

Bradley C. Wright

Christopher R. Glembocki

LGJ/BCW/CRG

Enclosures:

One disk copy of application (Word file and Visio file)
Four paper copies of application
Three paper copies of "Guidelines For Reviewing A Patent Application Draft"

EXHIBIT C

Microsoft® Facsimile Transmittal

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Phone:(206)882-8080
Telex:160520 MSFT BVUE
FAX:(206) 869-1327

To: Chris Glembocki
Company: Banner & Witcoff
Phone: (202) 508-9100
Fax: (202) 508-9299

From: Kymerie Schmidt
Phone: (425) 703-7132
Fax: (425) 703-6204

Date: August 26, 1997
Pages including this cover page: 2

REVIEWED

RECEIVED
AUG 2 6 1997
BANNER & WITCOFF LTD.

Comments: **CONFIDENTIAL FAX**

URGENT – Please deliver to Chris Glembocki ASAP.

RE: Ecenter Patent Applications
MS 97172.1/97192.1/100660.1

The following is Dave Plummer's clarification of the formula. Please call Jennifer Norris asap (425-703-5899) if you have any questions.

CONFIDENTIALITY STATEMENT

The information contained in this facsimile message is legally privileged and confidential information intended only for the use of the addressee listed on this coversheet. If the reader of this message is not the intended recipient, you are hereby notified that any dissemination, distribution or copy of this telecopy is strictly prohibited. If you have received this facsimile in error, please immediately notify us by telephone at the number listed on this coversheet and return the original message to us at the above address via the postal service. We will reimburse any costs you incur in notifying us and returning the message to us. Thank you.

Jennifer Norris (LCA)

425 936-9731

From: Dave Plummer
Sent: Monday, August 25, 1997 11:22 AM
To: Jennifer Norris (LCA); David Barnes
Cc: Kymerie Schmidt (LCA); Neil Calvin (LCA); Joseph Matthews
Subject: RE: First Draft Patent Application - ECenter Architecture; UI Hardware Mode Switching; Previews in Folder View (MS#97172.1,97192.1 and 100660.1)

All content is designed for a fixed resolution (in this example, 640x480). The current resolutions horizontal size (in your example, 800) pixels is used to compute the ratio of the current resolution to the authoring resolution, hence:

800 / 640 = N
1.25 = N

This factor N is then applied to all fixed pixel counts and sizes, so that

N x 400 = 500

So, anything that was previously assumed to be 400 pixels when running at 640x480 is now forced up to 500 pixels when running at 800 (by 600) resolution.

- Dave

-----Original Message-----
From: Jennifer Norris (LCA)
Sent: Monday, August 25, 1997 10:15 AM
To: David Barnes; Dave Plummer
Cc: Kymerie Schmidt (LCA); Neil Calvin (LCA); Joseph Matthews
Subject: RE: First Draft Patent Application - ECenter Architecture; UI Hardware Mode Switching; Previews in Folder View (MS# 97172.1,97192.1 and 100660.1)

Dave and David -
We are getting close to our deadline for filing this application and I need your help on the remaining questions embedded in the draft:

P.26

800/640 x 400 = 500 pixels high on a large screen.

[NEED TO CLARIFY THIS FORMULA]

Top of Page 28:

Please provide additional information on how the different window sizes are specified]

<< File: ECENTER3.doc >>

I also need any additional comments you may have on this draft. please try to get this information to me by end of day today so that we can get the changes made and the application filed by Wednesday.

Thanks!
Jennifer Norris
Patent Group
Law and Corporate Affairs

-----Original Message-----
From: Jennifer Norris (LCA)
Sent: [redacted]
To: David Barnes; Dave Plummer
Cc: Kymerie Schmidt (LCA); Neil Calvin (LCA); Joseph Matthews
Subject: RE: First Draft Patent Application - ECenter Architecture; UI Hardware Mode Switching; Previews in Folder View (MS# 97172.1,97192.1 and 100660.1)